Tim Hartnell

El libro gigante de los juegos para ordenador

ZX Spectrum/VIC-20/Commodore-64/Apple II/Dragon 32
Oric/IBM pc

LIBRO GIGANTE DE LOS JUEGOS PARA ORDENADOR

LIBRO GIGANTE DE LOS JUEGOS PARA ORDENADOR

Tim Hartnell

ANAYA MULTIMEDIA

MICROINFORMATICA

Título de la obra original:
GIANT BOOK OF COMPUTER GAMES

Traducción de: Pilar Vázquez
Diseño de colección: Antonio Lax

Primera edición, primera reimpresión

Reservados todos los derechos. Ni la totalidad ni parte de este libro puede reproducirse o transmitirse por ningún procedimiento electrónico o mecánico, incluyendo fotocopia, grabación magnética o cualquier almacenamiento de información y sistema de recuperación, sin permiso escrito de Ediciones Anaya-Multimedia, S. A.

First published in Great Britain by:
Interface Publications 1983

Copyright © Tim Hartnell, 1983

© EDICIONES ANAYA MULTIMEDIA, S.A., 1984
Villafranca, 22. 28028 Madrid
Depósito legal: S. 711-1984
ISBN: 84-7614-002-9
Printed in Spain
Imprime: Gráficas Ortega, S.A.
Polígono El Montalvo - Salamanca, 1984

INDICE

INTRODUCCION ..... *Pág.* 9
Cómo ejecutar los programas 10

JUEGOS DE MESA 15
Cómo juega el ordenador 17
AJEDREZ 21
GOMOKU 44
CUATRO EN RAYA 49
DAMAS 56
SHOGUN 65
AWARI 69
KNIGHTSBRIDGE 76
REVERSI/OTHELLO 83

AVENTURA 93
LA FORTALEZA DE LOS ENANOS 100
EL DUQUE DE DRAGONFEAR 106
EL LEGADO BANNOCHBURN 116

SIMULACIONES 139
LA SEÑORA XENOFOBIA 141
PRACTICAS RURALES 146
PRESIDENTE DEL CONSEJO DE ADMINISTRACION . 151

JUEGOS DE DADOS 161
CHEMIN DE ORDENADOR 163
SIETE/ONCE 171
SIN SUDOR 174
UNO Y VEINTE 179

OJOS DE SERPIENTES 184
POR ENCIMA Y POR DEBAJO 190

INTELIGENCIA ARTIFICIAL 197
ELIZA 201
CEREBRO ELECTRONICO 207
PROBOSCIDEOS 214
AMANUENSE 221

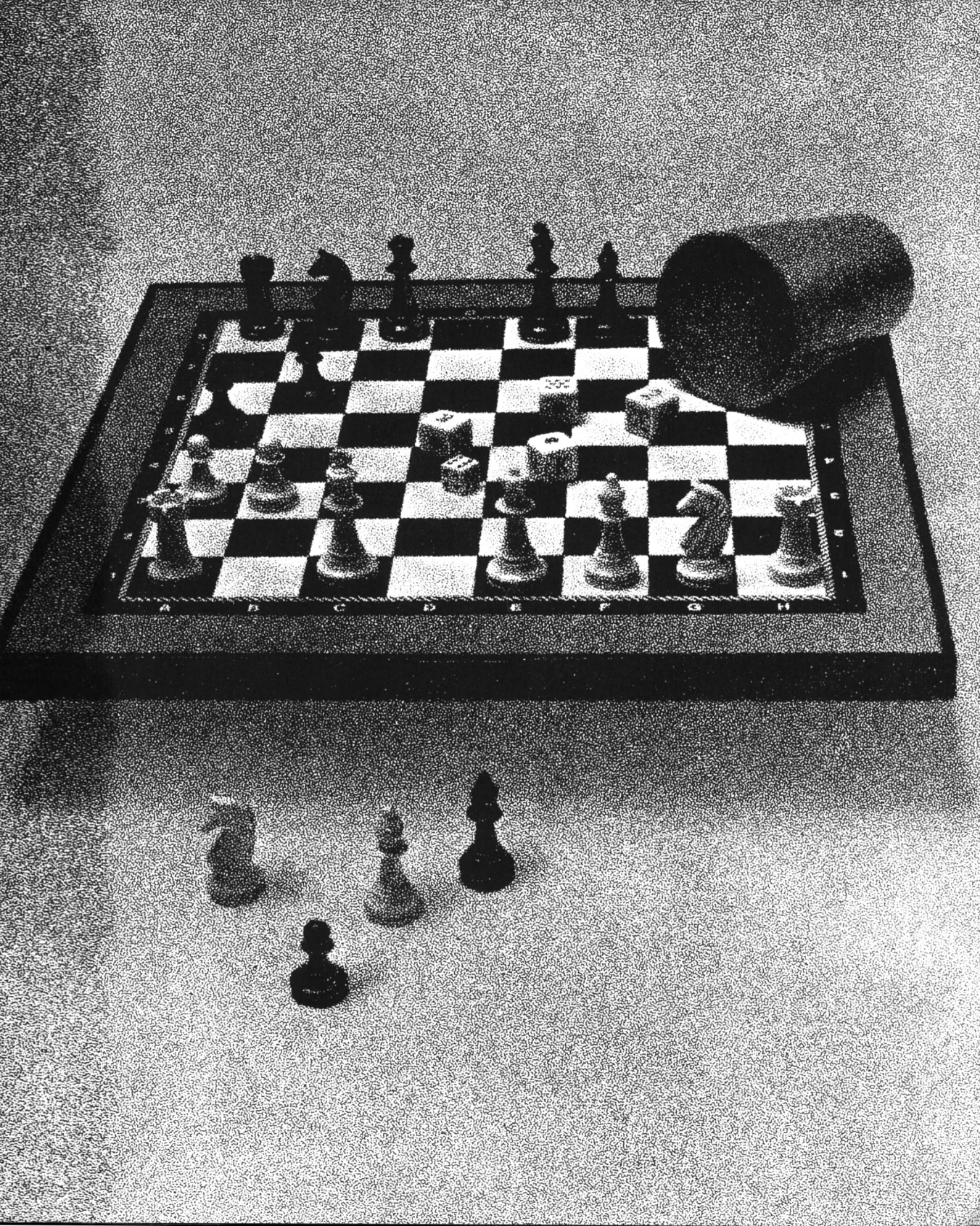
SOLO PARA DIVERTIRSE 227
RESORTE OCULTO 228
NIMGRAB 231
CADDY 234
CABALGATA DE PUZZLES 239
CAMPO MINADO 253
CEROS Y CRUCES 259
LAS VEGAS 264

Diversión con la impresora 271
CELESTIA 271
VALLA PUBLICITARIA 277

JUEGOS ESPACIALES 281
ALUNIZAJE I y II 282
SUPERGUERRA 287

JUEGOS MENTALES 297
FASTERMIND 298
ESTRELLAS DE IDAHO 301
EJECUTOR 306
CUBISTA 310
MUMBLE MARBLE 314
CICLOTRON 317
SWITCHEROO 321
FLIPPER 324
EN BUSCA DE DARYL 328

PARA CREAR JUEGOS PROPIOS 343
JUEGOS PARA ADAPTAR 344
BIBLIOGRAFIA 352
GLOSARIO 355


A B C D E F G H

INTRODUCCION

Erase una vez una época en la que escribir programas para jugar con el ordenador no era cosa fácil. Allá en los oscuros y confusos días de nuestro pasado se podían llegar a pagar entre 80 y 100 dólares por una sola hora de acceso a una terminal y esto no era lo peor: la única salida con la que contaban nuestros abnegados antepasados era un ruidoso y casi ininteligible teleimpresor. Imagínese lo que le hubieran supuesto estas limitaciones a alguien como usted, que seguro que puede llegar a pasarse horas y más horas mejorando, puliendo con todo amor, su última obra maestra.

Afortunadamente nosotros ya no tenemos que sufrir esas condiciones medievales. El lujo de poseer un ordenador propio, que utiliza una pantalla de televisión, con un acceso ilimitado a la máquina, le permite a uno trabajar cuanto tiempo quiera creando sus propios juegos y programas.

Este libro es una selección de mis juegos de ordenador favoritos. Espero que no pueda resistir la tentación de ejecutarlos en seguida y que por lo menos algunos de ellos pasen a formar parte de sus propios programas favoritos. De lo que estoy seguro es de que en las próximas semanas no le va a faltar diversión.

He procurado que los juegos fueran variados para acertar con todos los gustos. Como verá si echa un vistazo al libro, la selección incluye de todo. Desde los juegos de mesa «clásicos», como el AJEDREZ y el OTHELLO, a los programas de aventuras, como LA FORTALEZA DE LOS ENANOS y EL LEGADO DE BANNOCHBURN, pasando por aquellos del tipo de LA SEÑORITA DE XENOFOBIA, que no son sino simples simulaciones y cuyo único objetivo es el de divertirse un rato. Este libro contiene más de cuarenta juegos, todos ellos escritos con la intención de que su ordenador no vuelva a tener la oportunidad de quejarse de «aburrimiento».

He escrito un libro deliberadamente abierto. Es decir, en contraste con muchas de las otras colecciones de programas para juegos existentes en el mercado, yo he partido de la base de que cada lector querrá adaptar y mejorar el programa una vez que lo haya ejecutado. Esta es la razón por la que he incluido interrupciones en muchos de ellos, en algunos casos línea a línea, con el fin de que uno pueda darse cuenta de qué función tiene cada sección de código. Asimismo, en la introducción a cada apartado he hecho un pequeño comentario sobre la filosofía que entraña la escritura de ese determinado tipo de juegos, al tiempo qué he señalado cuáles son los algoritmos básicos que se pueden usar para el mismo y, en general, he intentado proporcionarle al lector una base sólida sobre la que desarrollar sus propias técnicas en la escritura de juegos de ordenador.

Al final del libro el lector encontrará una sección dedicada, casi por entero, a proporcionarle ideas que se pueden convertir en programas para jugar con el ordenador.

COMO EJECUTAR LOS PROGRAMAS

Para escribir este libro utilicé, de un modo totalmente deliberado, la versión de BASIC más general que me fue posible. Así pues, no aparecerán ni PEEKS ni POKES, ni se usarán conjuntos de caracteres gráficos ni instrucciones del tipo de SOUND o PLAY. Parto del supuesto de que mis lectores tendrán acceso a READ y DATA y de que sus pantallas tendrán entre treinta y dos y cuarenta caracteres de ancho (en el caso de tener un ordenador del tipo VIC-20 cuya pantalla no es tan ancha, en seguida se dará cuenta de que es extremadamente sencillo adaptar la salida del programa para que sirva para un ancho, pongamos, de doce caracteres, ya que la mayoría de los juegos sólo necesitan el ancho de un tablero de juego). Se utilizan las órdenes estándar para el manejo de series (como LEFT, RIGHT, y MID), las cuales han de traspasarse al mecanismo de segmentación de series de Sinclair en el caso de tener un ordenador del tipo T/S 2000 o Spectrum. Esto no entraña ninguna dificultad e incluso he procurado que haya que utilizar esas órdenes lo menos posible, de modo que la mayoría de los programas debería pasar a su ordenador sólo con escribirlos directamente.

* N. del E. Muchos de los mensajes de los programas utilizan la letra Ñ. En los listados del libro no se ha utilizado para darles carácter más general. El lector deberá adaptar estos mensajes según el procedimiento de su ordenador para generar la Ñ.

Por supuesto, tendrá que jugar un poco con la presentación en pantalla, para hacerlo lo más efectivo posible. Y ya que toco este tema, espero que cada lector modifique y adapte los programas de modo que pueda sacar el máximo partido al sistema que utilice, añadiéndole en donde se pueda sonido y color, además de los gráficos propios del sistema.

Muchos de los programas vuelven a dibujar el tablero, o campo de juego, después de cada jugada. Por razones de simplificación, he precedido de una orden CLS para borrar la pantalla cada una de estas nuevas reimpresiones. Algunos sistemas exigen una orden del tipo PRINT «CLR», de modo que en el caso de los usuarios de esos sistemas se ha de hacer esta sustitución cada vez que se dé el caso. Y lo que es más importante, en seguida observará que después de que la pantalla ha quedado borrada al principio del juego, la salida del programa puede hacerse mucho más efectiva si se sustituye CLS por una orden del tipo «home» con el fin de que el programa se imprima de cada vez superpuesto sobre sí mismo. En un juego como EL CAMPO MINADO, por ejemplo, esto da la impresión de que los robots se mueven. Lo mismo puede apreciarse en los juegos de mesa. De hecho, el ver cómo las fichas se mueven en el tablero es de lo más efectista y añade un placer más a la ejecución del programa.

Así pues, si su sistema cuenta con los medios para reajustar el cursor en la parte superior de la pantalla sin tener que borrarla primero (del tipo de PRINT AT 0,0 o PRINT@ o LOCATE 1,1 u otros del mismo tipo) asegúrese de que pone cualquiera de ellos en vez del CLS al inicio de la sección de reimpresión.

Estos programas fueron escritos en un IBM PC. En este ordenador, si uno quiere generar números enteros aleatorios entre el 1 y el 10, ha de utilizar una orden del tipo A = INT (RND*10) + 1. Como comprobará éste es el formato que se utiliza a lo largo de todo el libro. Si su sistema no lo acepta, tendrá que sustituir ya sea A= RND(10) o A = INT(RND (0) * 10) + 1 o A = INT(RND(1)*10) +1 (y es posible que tenga que preceder esta orden de LET). Lo más seguro es que el lector ya sepa cómo se realizan estas operaciones, pero si le quedara alguna duda, lo mejor es que consulte al apartado RND o «números aleatorios» en el manual de uso de su sistema.

El ordenador en el que yo desarrollé estos programas da siempre la misma secuencia de números aleatorios cada vez que se ejecuta el programa. Como los números que están más o menos distribuidos aleatoriamente se necesitan en muchos programas, lo que he hecho ha

sido inicializar, sembrar, en cada caso, el generador de números aleatorios siguiendo uno de los métodos que se describen a continuación.

En unos casos he utilizado INKEY$ para hacer un bucle en el programa aumentando una variable al hacerlo hasta el momento de presionar una clave, tras lo cual he utilizado esa variable para inicializar el generador de números aleatorios (como en RANDOMIZE N); en otros, he tomado como «semilla» para inicializar o sembrar los segundos del reloj incorporado al ordenador (lo que da lugar en muchos programas a esa aterradora línea RANDOMIZE VAL (RIGHT$ (TIME$, 2)).

Si se dispone de un ordenador que no produce los mismos números en la misma secuencia cada vez que se ejecuta el programa, no hay más que saltarse esas líneas por las buenas. También puede, si su sistema lo acepta, utilizar simplemente RANDOMIZE (o RAND) sin acompañarlo de número alguno. El manual de uso indicará en cada caso lo que se debe hacer. Si de todas maneras no estuviera seguro de lo que tiene que hacer y su ordenador no aceptara esa larga línea que incluye TIME$, omítala toda. El programa funcionará a la perfección sin ella.

Aunque gran parte de la salida entrecomillada en las sentencias PRINT está en minúsculas, todos los programas (a excepción del ajedrez que necesita las minúsculas para significar las fichas del jugador humano), esperan la entrada en mayúsculas. Si su sistema no cuenta con minúsculas, introduzca simplemente el material de las sentencias PRINT en mayúsculas. Yo lo introduje en minúsculas porque me pareció que así quedaba más bonito, pero no tiene nada que ver con la ejecución real del programa.

De hecho, verá como cuando empiece a introducir el programa, usted mismo hará automáticamente todos los pequeños cambios precisos para acomodarlo a las características peculiares del sistema que utilice. Si he hecho todas estas aclaraciones, ha sido sólo para que el lector no perdiera el tiempo antes de empezar a ejecutar los programas, pero lo más seguro es que estén de más y que ninguno de mis lectores tenga que hacer uso de ellas. Conque no se deje intimidar por la extensión de esta introducción y siga adelante.

La mayoría de los programas incluidos en este libro entran bien en 8K. La excepción es EL LEGADO BANNOCHBURN que exige casi 17K en mi sistema. El ajedrez ocupa 8K y los programas más cortos vienen a hacerlo por debajo de 4K. Si tiene dificultad en introducir un programa en su sistema debido a escasez de memoria, suprima las

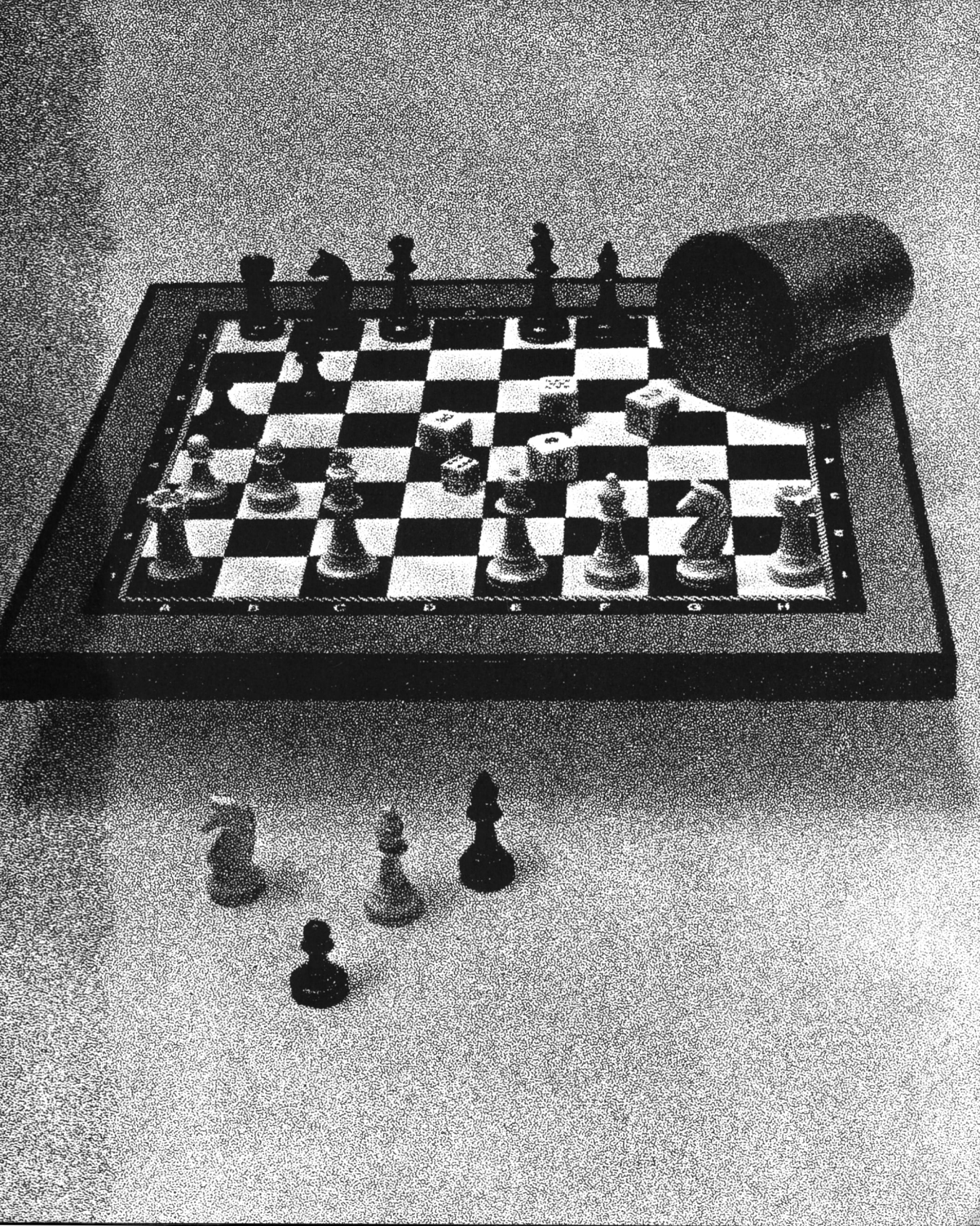
sentencias REM (la mayoría de las llamadas GOSUB y GOTO no se refieren al REM principal sino que van a la propia rutina que sigue al REM identificador) y acorte las sentencias PRINT. Es imposible predecir la memoria que necesitará cada programa en los diferentes sistemas, ya que ésta puede estar organizada de varias maneras diferentes y el espacio de trabajo que el programa requiere varía de un sistema al otro.

Para que los programas le quepan en su máquina, si no dispone de mucha memoria (y para que el tiempo de introducción sea el menor) muy pocos programas incluyen instrucciones de juego. Estas están en el libro. No obstante, si cuenta con la memoria necesaria (y tiene costumbre de hacerlo) puede añadir sin dificultad una línea "¿Desea instrucciones?" como parte de la subrutina de inicialización incluyendo después una explicación abreviada sacada del texto del programa.

He seguido técnicas de programación estructuradas en todos los programas que se prestaran a este enfoque. Estas técnicas las explico en el material que acompaña al programa de ajedrez. Es posible que el lector encuentre, como me sucedió a mí mismo cuando al fin me decidí a poner en práctica todos los consejos que había leído en libros y artículos, que el trabajar «de arriba a abajo» hace mucho más sencilla y más precisa la tarea de escribir y modificar un programa. Además, es mucho más fácil llegar a terminar y ejecutar un programa cuando se ha escrito siguiendo esta pauta. Y lo que es todavía más importante: podrá llegar a las partes más divertidas —la propia ejecución— mucho antes que en el caso contrario.

Muchos programas incluyen un bucle de retardo. Como el retraso producido dependerá del sistema que se utilice, lo que hay que hacer es ajustar esos falsos bucles (que normalmente se mantienen en subrutinas hasta el final del programa) de modo que el funcionamiento sea lo más efectivo posible.

Finalmente unas palabras sobre las variables. En varios casos se han usado como variables palabras completas (como PUNTOS). Sin embargo, sólo las dos primeras letras son significativas. Si el sistema que se utiliza sólo acepta nombres variables de dos letras, introduzca las dos primeras (como PU en este caso). De aceptarlo el sistema, e incluso, aunque sólo reconozca las dos primeras letras de un nombre variable, se ha de incluir el nombre entero. Haciendo esto, se dará cuenta de que es mucho más fácil saber lo que está sucediendo en el programa en cada momento.


A B C D E F G H

JUEGOS DE MESA

Ciertos juegos no hubieran existido nunca sin los ordenadores. Piense en todas las variedades de Invasores del Espacio o similares. Ninguno de ellos hubiera podido existir antes de que se desarrollara el ordenador y el uso de la televisión como pantalla de salida. Muchos otros juegos, sin embargo, tenían miles de años de existencia cuando se empezaron a desarrollar los ordenadores, lo que no impide que cada vez sean más los juegos tradicionales, de mesa fundamentalmente, que se están informatizando. Es más, los juegos de mesa son ahora tanto o más apreciados por los aficionados a los juegos en el ordenador como lo eran en los días anteriores a la aparición de éste.

Este capítulo incluye una generosa selección de juegos:

AJEDREZ
GOMUKU
CUATRO EN RAYA
DAMAS
SHOGUN (HASAMI SHOGI)
AWARI
KNIGHTSBRIDGE
REVERSI/OTHELLO

Empezaremos haciendo un breve comentario sobre la manera de modificar los juegos de mesa o tablero para que el ordenador pueda manejarlos. Las ideas que comentaré en las próximas páginas se han aplicado a todos los juegos de mesa que aparecen en este capítulo, salvo el Awari.

Todos los programas incluidos en este capítulo tienen una característica común. Una característica de la que se puede servir el lector a la hora de desarrollar programas para sus juegos de mesa favoritos.

En primer lugar observe este diagrama. Es un tablero de ajedrez numerado de modo que el ordenador lo puede manejar con facilidad.

	1	2	3	4	5	6	7	8
8	81	82	83	84	85	86	87	88
7	71	72	73	74	75	76	77	78
6	61	62	63	64	65	66	67	68
5	51	52	53	54	55	56	57	58
4	41	42	43	44	45	46	47	48
3	31	32	33	34	35	36	37	
2	21	22	23	24	25	26	27	28
1	11	12	13	14	15	16	17	18

Puede indicarle cualquier casilla del tablero refiriéndose a uno de los números situados en el margen izquierdo de éste (como el 3) y después a uno de los situados en la parte superior (como el 4). En este caso, la línea numerada 3 (partiendo de los números situados a la izquierda del tablero) y la línea numerada 4 (partiendo de los números situados en la parte superior) se encuentran en la casilla 34. Si quiere mover una ficha, puede hacerlo introduciendo el número de la casilla desde la que hace el movimiento (pongamos la 55) y después el número de la casilla hacia la que se mueve (pongamos la 66), el ordenador entenderá perfectamente lo que está haciendo. No es necesario cambiar los números introducidos por el jugador humano por otro conjunto diferente con el fin de que el ordenador los interprete.

Este es el primer secreto cuando se están escribiendo programas de juegos de mesa para jugar con el ordenador. El segundo es que un tablero de este tipo presenta otra gran ventaja frente a los que están simplemente numerados en orden de 1 a 64. Esta ventaja consiste en que con éste al mover una ficha en cualquier dirección —independientemente de dónde se empiece— la diferencia entre las casillas es siempre la misma.

Intentaré explicar este concepto ya que puede parecer un poco críptico. Si se avanza una casilla hacia arriba y hacia la derecha —como la jugada de una ficha de ajedrez— se moverá, por ejemplo, de la 24 a la 35, o de la 53 a la 64, o de la 71 a la 82. Pero observe que, independienemente de cuál sea su posición en el tablero, la diferencia entre la casilla de origen y la casilla a la que se mueve es siempre 11. Si avanza en diagonal hacia la izquierda, se moverá, por ejemplo, de la 26 a la 35 (más 9), o de la 66 a la 75 (también más 9) o de la 22 a la 31 (otra vez más 9).

Esta previsibilidad hace que sea relativamente sencillo crear un tablero que el ordenador sea capaz de manejar.

COMO JUEGA EL ORDENADOR

Imagine que el ordenador tiene una ficha en la casilla 24. Podría programarse para que compruebe todas las casillas del tablero y cada vez que encuentre alguna de sus fichas verifique si hay una ficha del jugador humano en una casilla con el mismo número que la casilla en la que él tiene su ficha (es decir, el 24 en este caso), más 11 (es decir, en la 35); y que compruebe además si la casilla 46 (es decir, 35 más 11)

está vacía. Si todas estas condiciones se cumplen, el ordenador puede mover su ficha de la casilla 35 a la 46 comiéndose así la ficha del jugador humano que ocupa la casilla 35.

Así es, en esencia, cómo funcionan la mayoría de los juegos de mesa en el ordenador —de las damas al ajedrez, pasando por el Othello— basados en una sencilla cuadrícula de 8 por 8 numerada de la manera que hemos visto.

Si fuera usted a escribir un programa de ajedrez basado en este tablero (como, de hecho, lo está el que veremos a continuación), podría especificar las jugadas del caballo, por ejemplo, al saber que éste siempre puede avanzar a las casillas que están, a partir de la que ocupa en ese momento el caballo, a la siguiente «distancia numérica»:

21 12 −8 −19 −21 −12 19 8

Inténtelo usted ahora. Poniendo una moneda en la casilla número 55 y moviéndola como si se tratara del caballo, calcule la relación matemática entre la casilla de partida y la casilla a la que va a avanzar. Verá cómo las diferencias son las mismas que las señaladas más arriba.

Los programas de este capítulo, y los de muchos otros capítulos del libro, fueron escritos siguiendo un esquema «de arriba a abajo» en un intento de que salieran unos programas estructurados y relativamente fáciles de seguir y modificar. Muchos de los programas los escribí introduciendo al principio el siguiente «código puro»:

```
10 REM NOMBRE DEL JUEGO
20 GOSUB 9000: REM INICIALIZACION
30 GOSUB 8000: REM DIBUJO DEL TABLERO
40 GOSUB 1000: REM JUGADA DEL ORDENADOR
50 GOSUB 8000: REM DIBUJO DEL TABLERO
60 GOSUB 7000: REM JUGADA DEL HUMANO
70 Comprobación de si el ordenador o el humano han ganado, en
   ese caso, final del juego.
80 GOTO 30
```

En casi todos los casos tenía ya escrita esta estructura antes de tener la menor idea de cómo iba a alcanzar las metas que yo mismo me había marcado para cada una de las subrutinas.

Todos los programas fueron escritos primeramente en papel antes de poner un dedo en el ordenador, de modo que se podía «ejecutar a

mano» una versión del juego antes de empezar a vérselas propiamente con éste en el ordenador. De este modo se pueden detectar y corregir justo al principio los peores errores.

Recomiendo fervientemente al lector que siga un procedimiento similar. Yo mismo había leído este consejo en varios artículos y libros y, como se puede suponer, lo había pasado por alto hasta que en una ocasión, pasando unos días de vacaciones en Gales, a kilómetros de distancia del ordenador más próximo, me entraron unas ganas locas de escribir un programa de ajedrez. El programa que aparece en este libro está basado en aquel programa original, ideado durante mis largos paseos por la playa en compañía de mi perro y puesto por escrito en una granjita alquilada de la costa galesa.

Entre las muchas ventajas que descubrí al programar primero en papel, hay una fundamental que es lo poco que te importa deshacerte de bloques completos de código si te das cuenta de que no funcionan. Es mucho más difícil decidirse a borrar una sección completa de código de un programa una vez que éste ya está introducido en el ordenador, de lo que puede serlo el hacer trocitos una hoja de papel y tirarlos a la papelera. La tentación que le suele asaltar a uno –cuando las líneas del programa ya están en el ordenador— es la de juguetear con ellas intentando que funcione, aunque no sea exactamente como se esperaba.

El trabajar primero en papel tiende, pues, a evitar el tener que andar haciendo un remiendo tras otro para conseguir que funcione un código que, en realidad, no debería estar en ese programa. Por otro lado, el trabajar siguiendo un esquema estructurado, como el que he descrito, simplifica la labor, pues se sabe qué partes del programa realizan cada tarea. En el esquema citado, yo sé, por ejemplo, que el tablero es impreso por la subrutina que empieza en la línea 8000. Así pues, si el tablero sale con un aspecto un poco extraño, ya sé inmediatamente en qué sección del código he de concentrarme.

Además de ser de una gran ayuda en un primer momento a la hora de ejecutarlo, un programa estructurado invita a hacerle mejoras. Por ejemplo, suponga que ya cuenta con un programa de ajedrez propio y dispuesto para ser ejecutado. Si en algún momento quisiera volver sobre él para intentar mejorar su juego, le bastaría con sólo revisar la parte del código que cubre el juego del ordenador. No tendría que recorrer kilómetros de código calculando cuál es la función de cada línea y cuáles son las líneas que controlan el juego del ordenador.

Los métodos que acabo de apuntar son los que yo he usado en esta parte del libro y en casi todas las demás. Puede ser instructivo echar un vistazo a algunos de los programas y calcular mediante los listados qué sección realiza cada tarea. Verá que en muchos programas he incluido líneas de asteriscos a modo de sentencias REM. Estos separan el programa en módulos independientes, lo cual supone una gran ayuda para seguirlo sin dificultad.

AJEDREZ

El ajedrez es uno de los grandes desafíos a los que puede enfrentarse un programador de ordenadores. Analizar el modo en que los humanos juegan al ajedrez, desmenuzar posteriormente ese análisis en un número de ideas lo suficientemente claras como para poder traspasarlas a un programa es una tarea formidable.

Este programa de ajedrez no pretende ser un intento particularmente logrado de superar el desafío que acabo de describir. Aunque con este programa el ordenador jugará al ajedrez con una base razonablemente coherente, no se puede decir que juegue extraordinariamente bien y desde luego el intentar vencerlo no supone un desafío real. ¿Por qué lo he incluido, entonces, en este libro junto a otros juegos en los cuales el ordenador se muestra prácticamente invencible?

La razón de incluirlo ha sido precisamente el hecho de que el ajedrez cuente con innumerables atractivos para los programadores de ordenador. Muy pocos programas de ajedrez se han publicado hasta el momento (conozco uno que apareció en el número de diciembre de 1981, de la revista *Creative Computing,* bajo el título de «Chess C.4» y cuyo autor era Michael Rakaska; dos más que aparecieron en libros; y el listado del Sargon II), lo que demuestra la dificultad que entraña la producción de un programa de este tipo. Me pareció que la propia escasez de programas de ajedrez publicados era un argumento de peso en favor de incluir éste.

Durante cientos de años la mente humana no ha dejado de acariciar la posibilidad de crear una máquina que fuera capaz de jugar al ajedrez frente a un contrincante humano. En el último tercio del siglo

XVIII el Barón von Kempelen construyó un autómata jugador de ajedrez con el que hizo su primera exhibición en Viena en el año 1770 ante la familia real; la Europa de la época quedaría totalmente fascinada con esta idea. Lamentablemente, la máquina era poco más que un elaborado truco de malabarista, con un hombre, que operaba el mecanismo de juego, oculto en un compartimento. No obstante, el engaño estaba bien logrado y un examen detenido de las jugadas muestra que el hombre que se arrebujaba dentro del mecanismo sabía lo que se hacía. En cierta ocasión en la que investigaba yo la historia del juego en el Museo Británico, fui a dar con un libro del siglo pasado, *50 Games Played by the Chess Automaton,* que algún lector aplicado había ido anotando en los márgenes, y me quedé sorprendido al comprobar que los comentarios más severos —del tipo de «esta jugada sólo la haría un principiante»— estaban siempre dirigidos al jugador humano y nunca a la máquina.

Pero a pesar de su fama, la máquina, en realidad, no jugaba al ajedrez. El primer intento real de un mecanismo que jugara una partida completa de ajedrez (diferente de las máquinas que se construyeron a principios de siglo) data de 1949, cuando el 9 de marzo de ese año Claude Shannon, un investigador de los Bell Telephone Laboratories, presentó un trabajo llamado *Programming a Computer for Playing Chess* en un congreso en Nueva York.

Como muy bien subraya David Levy en su fascinante libro *Chess and Computers* (Computer Science Press, Inc., Potomac, Maryland, 1976), la importancia real del trabajo de Shannon radica, no en el hecho de que fuera el primero, sino en el de que «muchas de las ideas originales de Shannon pueden verse todavía en los programas que se realizan hoy».

Shannon decía en su trabajo que el número de jugadas posibles en una partida de 40 jugadas se aproximaba a 10 elevado a la cientoveinteava potencia, un número que pertenece al mismo orden que los átomos en el universo. Está claro que cualquier programa que intentara evaluar todas esas jugadas se extendería durante un largo tiempo (pongamos diez años elevados a la nonanteava potencia) antes de que el primer peón del rey hiciera una incursión hacia el centro.

El hacer que este programa pudiera jugar, y conseguir que lo hiciera razonablemente rápido, exigía, como estoy seguro de que el lector habrá apreciado, cierto número de compromisos. Cambié la evaluación en profundidad por ciertas ideas que si bien no eran más que aproximativas, al menos eran ingeniosas (como el programar el or-

denador para que moviera las fichas de tal modo que dominara el tablero el mayor tiempo posible; desplegar los caballos lo antes posible; comer con los peones, independientemente del hecho de que el peón se enfrentara a una captura inmediata, cuando fuera posible; mantener el rey lo más inmóvil posible, al menos durante los primeros momentos del juego, cuando parece que tendría sentido hacer una cobarde aproximación; y dotar a las fichas que cuentan con más movilidad —como la reina— de un mecanismo de toma de decisiones equilibrado aleatoriamente que asegurara que no iba a dar un salto hasta el otro lado del tablero cada vez que pudiera hacerlo).

En la composición del programa seguí una estructura clara y precisa; una estructura que garantizara que el juego del ordenador fuera lo más rápido posible. En realidad, esta estructura no resultó tan transparente como yo había esperado cuando escribí el programa. El gran número y la complejidad de las exigencias que el juego demandaba al ordenador cuando conseguí ejecutar el programa, significaba que mis planes de producir un programa claramente estructurado habían fracasado en cierto modo. Sin embargo, el programa juega extremadamente rápido y si no lo hace del todo bien, al menos es ocurrente y no dejará de divertir y de exasperar a sus contrincantes.

Al final del listado hay una sección en la que se le indica la manera de modificar el programa para que juegue contra sí mismo. La demostración de este programa es fascinante y estoy seguro de que no dejará de atraer la atención, especialmente en el caso de aquellos que se consideran expertos en el juego.

La variación de «auto-ajedrez» hace uso de la opción del intercambio de lados durante el juego; el cambio se realiza después de cada jugada. Cuando uno haga intercambio de lados, como puede hacerlo el lector si cree que al ordenador le vendría bien una ayudita, le dará a éste el lado en el que estaba jugando él; las fichas se intercambian como si estuvieran reflejadas en un espejo situado en el centro del tablero. Es decir, una reina al cambiarse de lado termina en la casilla coloreada correcta, y así sucesivamente. Pruebe a cambiar de lado varias veces durante el juego y verá cómo la operación de este mecanismo es bastante sencilla.

Unas palabras más antes de pasar a nuestro programa de demostración. Es preciso vigilar el computador para asegurarse de que no hace trampas. En realidad es muy raro que las haga, pero (a no ser que su contrincante se sienta particularmente condescendiente y le dé igual) si, por ejemplo, se deja dar jaque, el contrincante ha de tomarlo

como un signo de que el programa abandona. Asimismo, también se puede dar el caso de que el programa tenga dificultades para escaparse del jaque; en este caso, el contrincante podría mostrarse un poco más tolerante ya que normalmente el ordenador se las suele arreglar para hacerlo en la siguiente jugada. Por otro lado, el contrincante humano siempre puede considerar cualquiera de estos fallos como un reconocimiento de la superioridad de su juego.

Antes del listado del programa, me gustaría que el lector observara estas jugadas de una partida jugada por el ordenador.

```
    A B C D E F G H

8   T C A D R A C T   8
7   p p p p p p p p   7
6   . . . . . . . .   6
5   . . . . . . . .   5
4   . . . . . . . .   4
3   . . . . . . . .   3
2   p p p p p p p p   2
1   t c a d r a c t   1

    A B C D E F G H

UN MOMENTO, POR FAVOR
YO MUEVO DESDE C 7  A C 5
```

```
    A B C D E F G H

8   T C A D R A C T   8
7   p p . p p p p p   7
6   . . . . . . . .   6
5   . . p . . . . .   5
4   . . . . . . . .   4
3   . . . . . . . .   3
2   p p p p p p p p   2
1   t c a d r a c t   1

    A B C D E F G H
```

```
DESDE (LETRA,NUMERO)?E2
HASTA?E4
Seleccione J   para dar Jaque
           C   para Copia por impresora
           I   para Intercambiar lados
           T   para Terminar el juego
O bien RETURN para continuar?
```

```
          A B C D E F G H

        8 T C A D R A C T 8
        7 P P . P P . P P 7
        6 . . . . . P . . 6
        5 . . P . . . . . 5
        4 . . P . P . . . 4
        3 . . . . . . . . 3
        2 P P . P . P P P 2
        1 t c a d r a c t 1

          A B C D E F G H

UN MOMENTO, POR FAVOR

          A B C D E F G H

        8 T C A D R A C T 8
        7 P P . P . . P P 7
        6 . . . . . P . . 6
        5 . . P . P . . . 5
        4 . . P . P . . . 4
        3 . . . . . . . . 3
        2 P P . P . P P P 2
        1 t c a d r a c t 1

          A B C D E F G H


DESDE (LETRA,NUMERO)?H2
HASTA?H4
Seleccione J   para dar Jaque
           C   para Copia por impresora
           I   para Intercambiar lados
           T   para Terminar el juego
O bien Return para continuar?
```

```
   A B C D E F G H          A B C D E F G H

8  T C A D R A C T  8       8  T C A . R A C T  8
7  . . . P . . . .  7       7  . . D P . . . .  7
6  P P . . . P P P  6       6  . C . . . P P P  6
5  . . P C P . . .  5       5  P . P . P . . .  5
4  P . P . P . . P  4       4  P . P . P . D P  4
3  . . . . . . . .  3       3  . . . . . . . .  3
2  . P . P . P P .  2       2  . P . P . P P .  2
1  t . a d r a c t  1       1  t . a . r a c t  1

   A B C D E F G H          A B C D E F G H
```

```
DESDE (LETRA,NUMERO)?D1                DESDE (LETRA,NUMERO)?B6
HASTA?G4                               HASTA?A8
Seleccione J   para dar Jaque
           C   para Copia por impresora
           I   para Intercambiar lados
```

Y aquí tenemos el listado del juego del AJEDREZ:

```
   10 REM Ajedrez
   20 GOSUB 2970
   30 GOTO 60
   40 GOSUB 2580
   50 GOSUB 2820
   60 GOSUB 2580
   70 REM ******************************
   80 IF A$ = "T" THEN END
   90 IF A$ = "I" THEN PRINT "INTERCAMBIAMOS LAS P
IEZAS":GOSUB 3540:A$=""
  100 IF A$= "C"THEN GOSUB 3730
  110 REM ******************************
  120 FOR Z = 1 TO 16:F(Z) = 0:NEXT Z
  130 U = 0
```

```
140 PRINT "UN MOMENTO, POR FAVOR"
150 FOR D = 1 TO 64:IF M(S(D)) >= AN AND M(S(D))
 <= TN THEN U = U + 1:F(U) = S(D):IF M(S(D)) = RN
THEN SR = S(D)
160 NEXT D:IF U < 3 THEN GOTO 2230
170 D=100:GOTO 650
180 FOR D = 1 TO U: IF M(F(D)) = RN THEN F(D) =
F(U):F(U) = SR
190 NEXT D
200 D = INT(RND(-N) * 3)
210 IF A$= "J" THEN D = 0
220 IF D < U THEN D = D + 1
230 Z = F(D):GOSUB 280
240 IF MM = 1THEN GOSUB 2500:GOTO 40
250 IF D < U THEN 220
260 GOTO 2360
270 REM ******************************
280 IF M(Z)=DN THEN GOSUB 910
290 IF M(Z)=TN THEN GOSUB 1170
300 IF M(Z)=AN THEN GOSUB 1420
310 IF M(Z)=CN THEN GOSUB 1690
320 IF M(Z)=PN THEN GOSUB 2240
330 RETURN
340 REM ******************************
350 IF M(X) = 114 THEN MM = 0:D = D + 1:RETURN
360 IF X + 9 > 88 THEN380
370 IF M(X+9) < 85 AND M(X+9) > 64 AND RND(-N) <
 .96 THEN RETURN
380 IF X-11 < 11THEN 400
390 IF M(X-11) < 85 AND M(X-11) > 64 AND RND(-N)
 < .96 THEN RETURN
400 AD = 0
410 AY = 0
420 AX = X + D(AY + AD)
430 IF AX < 11 OR AX > 88 THEN 460
440 AP = M(AX)
450 IF AP = D OR AP = T AND RND(-N) > .8 OR AP =
 A AND RND(-N) > .5 THEN RETURN
460 AY = AY + 1
470 IF AY < 8 THEN 420
480 AD = AD + 7
490 IF AD < 56 THEN 410
500 AY = 1
```

```
510 AX = X + C(AY)
520 IF AX < 11 OR AX > 88 THEN 540
530 IF M(AX) = C THEN RETURN
540 AY = AY + 1
550 IF AY < 9 THEN 510
560 AY = 1
570 AX = X + R(AY)
580 IF AX < 11 OR AX > 88 THEN 600
590 IF (M(AX) = R OR M(AX) = P)AND RND(-N) > .1
THEN RETURN
600 AY = AY + 1
610 IF AY < 9 THEN 570
620 MM = 1
630 RETURN
640 REM*******************************
650 Z = SR
660 Y = 0
670 Y = Y + 1
680 X = Z + C(Y)
690 IF X < 11 OR X > 88 THEN 600
700 IF M(X) = C THEN 1870
710 IF Y < 8 THEN 670
720 REM*******************************
730 L = 0
740 Y = 1
750 X = Z + D(Y+L)
760 IF X < 11 OR X > 88 THEN 810
770 IF M(X) = A OR M(X) = D OR M(X) = T THEN 187
0
780 IF M(X) <> V THEN 810
790 Y = Y + 1
800 IF Y < 8 THEN 750
810 L = L + 7
820 IF L < 49 THEN 750
830 X = Z + 11
840 IF X > 88 THEN 860
850 IF M(X) = P THEN 1870
860 X = Z - 11
870 IF X < 11 THEN 180
880 IF M(X) = P THEN 1870
890 GOTO 180
900 REM*******************************
910 L = 0
```

```
 920 Y = 1
 930 X = Z + D(Y+L)
 940 IF X < 11 OR X > 88 THEN 1000
 950 IF M(X) = 42 OR M(X) >= AN AND M(X) <= TN TH
EN 1000
 960 IF M(X) >= A AND M(X) <= T THEN GOSUB 350:IF
 MM <> 1 THEN 1000
 970 IF MM = 1 THEN RETURN
 980 Y = Y + 1
 990 IF Y < 7 THEN 930
1000 L = L + 7
1010 IF L < 42 THEN 920
1020 RETURN
1030 REM*********************************
1040 L = 0
1050 Y = 1
1060 X = Z + D(Y+L)
1070 IF X < 11 OR X > 88 THEN 1130
1080 IF M(X) <> V THEN 1130
1090 IF RND(-N) > .5 THEN GOSUB 350:IF MM = 0 THE
N GOTO 1130
1100 IF MM = 1 THEN RETURN
1110 Y = Y + 1
1120 IF Y < 8 THEN GOTO 1060
1130 L = L + 7
1140 IF L < 49 THEN 1050
1150 RETURN
1160 REM*********************************
1170 L = 0
1180 Y = 1
1190 X = Z + T(Y+L)
1200 IF X < 11 OR X > 88 THEN 1260
1210 IF M(X) = 42 OR M(X) >= AN AND M(X) <= TN TH
EN 1690
1220 IF M(X) >= A AND M(X) <= T THEN GOSUB 350:IF
 MM = 0 THEN GOTO 1260
1230 IF MM = 1 THEN RETURN
1240 Y = Y + 1
1250 IF Y < 7 THEN 1190
1260 L = L + 7
1270 IF L < 21 THEN 1180
1280 RETURN
1290 REM*********************************
```

```
1300 L = 0
1310 Y = 1
1320 X = Z + T(Y+L)
1330 IF X < 11 OR X > 88 THEN 1390
1340 IF M(X) <> V THEN 1390
1350 IF RND(-N) < .1 THEN GOSUB 350
1360 IF MM = 1 THEN RETURN
1370 Y = Y + 1
1380 IF Y < 7 THEN 1320
1390 L = L + 7
1400 IF L < 21 THEN 1310
1410 RETURN
1420 IF M(Z) <> AN THEN RETURN
1430 L = 0
1440 Y = 1
1450 X = Z + A(Y+L)
1460 IF X < 11 OR X > 88 THEN 1520
1470 IF M(X) = 42 OR M(X) >= AN AND M(X) <= TN TH
EN 1520
1480 IF M(X) >= A AND M(X) <= T THEN GOSUB 350:IF
MM <> 1THEN 1520
1490 IF MM = 1 THEN RETURN
1500 Y = Y + 1
1510 IF Y < 7 THEN 1450
1520 L = L + 7
1530 IF L < 21 THEN 1440
1540 RETURN
1550 REM*********************************
1560 L = 0
1570 Y = 1
1580 X = Z + A(Y+L)
1590 IF X < 11 OR X > 88 THEN 1650
1600 IF M(X) <> V THEN 1650
1610 IF RND(-N) > .05 THEN GOSUB 350:IF MM <> 1 T
HEN 1650
1620 IF MM = 1 THEN RETURN
1630 Y = Y + 1
1640 IF Y < 7 THEN 1580
1650 L = L + 7
1660 IF L < 21 THEN 1570
1670 RETURN
1680 REM*********************************
1690 Y = 1
```

```
1700 X = Z + C(Y)
1710 IF X < 11 OR X > 88 THEN 1750
1720 IF M(X) = 42 THEN 1750
1730 IF M(X) >= A AND M(X) <= T THEN GOSUB 350
1740 IF MM = 1 THEN RETURN
1750 Y = Y + 1
1760 IF Y < 9 THEN 1700
1770 RETURN
1780  REM*******************************
1790 Y = 0
1800 X = Z + C(INT(RND(-N) * 8 + 1))
1810 IF X < 11 OR X > 88 THEN 1800
1820 IF M(X) = 42 THEN 1800
1830 Y = Y + 1
1840 IF M(X) = V THEN GOSUB 350
1850 IF MM = 1 OR Y > 20 THEN RETURN
1860 GOTO 1800
1870 YR = 1
1880 Z = SR
1890 X = Z + R(YR):X1 = X
1900 IF X < 11 OR X > 88 THEN 2200
1910 IF M(X) = 42 OR M(X) > 64 AND M(X) < 85 THEN
2200
1920 IF M(X) > 96 AND M(X) < 117 THEN 2200
1930 Z = X
1940 Y = 0
1950 Y = Y + 1
1960 X = Z + C(Y)
1970 IF X < 11 OR X > 88 THEN 1990
1980 IF M(X) = C THEN 2200
1990 IF Y < 8 THEN 1950
2000 REM*******************************
2010 L = 0
2020 Y = 1
2030 X = Z + D(Y+L)
2040 IF X < 11 OR X > 88 THEN 2090
2050 IF M(X) = A OR M(X) = D OR M(X) = T THEN 220
0
2060 IF M(X) <> V THEN 2090
2070 Y = Y + 1
2080 IF Y < 8 THEN 2030
2090 L = L + 7
2100 IF L < 49 THEN 2030
```

```
2110 X = Z + 11
2120 IF X > 88 THEN 2140
2130 IF M(X) = P THEN 2200
2140 X = Z - 11
2150 IF X < 11 THEN 2170
2160 IF M(X) = P THEN 2200
2170 X = X1:Z = SR
2180 MM = 1
2190 GOSUB 2500:GOTO 40
2200 YR = YR + 1
2210 Z = SR
2220 IF YR < 9 THEN 1880
2230 PRINT"       ME RINDO, CAMPEON!!!":END
2240 X = Z + 9
2250 IF M(X) >= A AND M(X) <= T THEN MM = 1:IF M(
X) = P AND RND(-N) < .2 THEN MM = 0
2260 IF MM = 1 THEN RETURN
2270 IF Z = 12 THEN RETURN
2280 X = Z - 11
2290 IF M(X) >= A AND M(X) <= T THEN MM = 1:IF M(
X) = P AND RND(-N) < .2 THEN MM = 0
2300 RETURN
2310 REM*******************************
2320 IF Z - 10 * (INT(Z/10)) = 7 AND M(Z-1) = V A
ND M(Z-2) = V AND (M(Z-13) = V OR M(Z-13) = 42) AN
D (M(Z+7) = V OR M(Z+7) = 42) THEN X = Z - 2:MM =
1:RETURN
2330 IF M(Z-1) = V AND M(Z-12) < 97 AND M(Z+8) <
97 THEN X = Z - 1:MM = 1:RETURN
2340 IF RND(-N) < .05 AND M(Z-1) = V THEN X = Z -
1:MM = 1
2350 RETURN
2360 D = INT(RND(-N) * RND(-N) * 5):IF D > U THEN
GOTO 2360
2370 IF D < U THEN D = D + 1
2380 Z = F(D)
2390 IF M(Z) = PN THEN GOSUB 2320
2400 IF M(Z) = CN THEN GOSUB 1790
2410 IF M(Z) = AN THEN GOSUB 1560
2420 IF M(Z) = TN THEN GOSUB 1300
2430 IF M(Z) = DN THEN GOSUB 1040
2440 IF M(Z) = RN AND A$ <> "J" AND RND(-N) < .07
THEN GOSUB 1870
```

```
2450 IFMM = 0 AND D < U THEN GOTO 2370
2460 IF MM = 1 THEN GOSUB 2500:GOTO 40
2470 UK = UK + 1:IF UK > 8 THEN 2230
2480 GOTO 2360
2490 REM*********************************
2500 IF M(Z) = RN AND A$<> "J" AND RND(-N) > .1 T
HEN MM = 0:GOTO 2360
2510 IF M(Z) = PN AND ((X - 10 * INT(X / 10) > Z
- 10 * INT(Z / 10) OR ABS(X - Z) > 11)) THEN MM =
0:U = U + 1:GOTO 230
2520 IF M(X) = R THEN PRINT " JAQUE":MM = 0:U = U
+ 1:GOTO 230
2530 M(X) = M(Z):M(Z) = V
2540 PRINT "YO MUEVO DESDE ";
2550 FZ = INT(Z / 10):PRINT CHR$(FZ + 64);Z - 10
* FZ ;" A ";: FX = INT(X / 10):PRINT CHR$(FX + 64)
;X - 10 * FX:FOR O = 1 TO 1000:NEXT O
2560 RETURN
2570 REM*********************************
2580 CLS
2590 GOSUB 2670
2600 FOR X = 8 TO 1 STEP-1
2610 PRINT TAB(10);X;" ";
2620 FOR Y = 10 TO 80 STEP 10
2630 IF M(Y+1) = PN THEN M(Y+1) = DN
2640 IF M(Y+8) = P  THEN M(Y+8) = D
2650 PRINT CHR$(M(X+Y));" ";
2660 NEXT Y:PRINT X:NEXT X:MM = 0
2670 PRINT:PRINT TAB(14);"A B C D E F G H":PRINT
2680 REM*******************************
2690 RETURN
2700 Z = SR
2710 QK = 0
2720 M = Z + R(QK)
2730 IF M < 11 OR M > 88 THEN 2780
2740 IF M(M) = 42 OR M(M) > 64 AND M(M) < 85 OR M
M = 0 THEN 2780
2750 X = M
2760 SR = Z
2770 RETURN
2780 IF QK < 8 THEN 2720
2790 IF A$ <> "J" THEN RETURN
2800 GOTO 2230
```

```
 2810 REM*****************************
 2820 PRINT
 2830 INPUT "DESDE (LETRA,NUMERO)";A$
 2840 IF LEN(A$) <> 2 THEN 2820
 2850 INPUT "HASTA";B$
 2860 IF LEN(B$) <> 2 THEN 2850
 2870 X = 10*(ASC(A$) - 64) + VAL(RIGHT$(A$,1))
 2880 Y = 10*(ASC(B$) - 64) + VAL(RIGHT$(B$,1))
 2890 PRINT "Seleccione J   para dar Jaque"
 2900 PRINT "           C   para Copia por impreso
ra"
 2910 PRINT "           I   para Intercambiar lado
s"
 2920 PRINT "           T   para Terminar el juego
"
 2930 INPUT "O bien RETURN para continuar";A$
 2940 IF M(Y) >= 65 AND M(Y) <= 84 AND M(Y) <> 67
THEN GOSUB 3660
 2950 M(Y) = M(X):M(X) = 46:RETURN
 2960 REM*****************************
 2970 CLS:PRINT "POR FAVOR,PONGA EL TECLADO EN MAY
USCULAS":PRINT"Y A CONTINUACION PULSE RETURN"
 2980 N=N+1:IF INKEY$="" THEN2980
 2990 RANDOMIZE:CLS:PRINT"UN MOMENTO, POR FAVOR"
 3000 DEFINT A - Z
 3010 MM = 0:A$ = ""
 3020 DIM M(99),T(28),A(28),C(8),D(56),R(8),Z(88),
S(64),F(16)
 3030 P = 112:T = 116:A = 97:C = 99:D = 100:R = 11
4:V = 46
 3040 PN = 80:TN = 84:AN = 65:CN = 67:DN = 68:RN =
 82
 3050 FOR Z = 1 TO 99:M(Z) = -99:NEXT Z
 3060 REM*****************************
 3070 FOR Z = 1 TO 64:READ X:READ Y:M(X) = Y:NEXT
Z
 3080 DATA 18,84,28,67,38,65,48,68
 3090 DATA 58,82,68,65,78,67,88,84
 3100 DATA 17,80,27,80,37,80,47,80
 3110 DATA 57,80,67,80,77,80,87,80
 3120 DATA 16,46,26,46,36,46,46,46
 3130 DATA 56,46,66,46,76,46,86,46
 3140 DATA 15,46,25,46,35,46,45,46
```

```
3150 DATA 55,46,65,46,75,46,85,46
3160 DATA 14,46,24,46,34,46,44,46
3170 DATA 54,46,64,46,74,46,84,46
3180 DATA 13,46,23,46,33,46,43,46
3190 DATA 53,46,63,46,73,46,83,46
3200 DATA 12,112,22,112,32,112,42,112
3210 DATA 52,112,62,112,72,112,82,112
3220 DATA 11,116,21,99,31,97,41,100
3230 DATA 51,114,61,97,71,99,81,116
3240 REM*****************************
3250 RESTORE 3270
3260 FOR Z = 1 TO 8:READ C(Z):NEXT Z
3270 DATA 19,-19,21,-21,-8,8,12,-12
3280 FOR Z = 1 TO 28:READ T(Z):NEXT Z
3290 DATA 10,20,30,40,50,50,50
3300 DATA -1,-2,-3,-4,-5,-5,-5
3310 DATA -10,-20,-30,-40,-50,-50,-50
3320 DATA 1,2,3,4,5,5,5
3330 RESTORE 3350
3340 FOR Z = 1 TO 28:READ A(Z):NEXT Z
3350 DATA -11,-22,-33,-44,-55,-55,-55
3360 DATA 11,22,33,44,55,55,55
3370 DATA 9,18,27,36,45,45,45
3380 DATA   -9,-18,-27,-36,-45,-45,-45
3390 RESTORE 3290
3400 FOR Z = 1 TO 56:READ D(Z):NEXT Z
3410 FOR Z = 1 TO 8:READ R(Z):NEXT Z
3420 DATA 1,11,9,10,-10,-9,-11,-1
3430 FOR Z = 1 TO 64:READ S(Z):NEXT Z
3440 DATA 46,56,36,66,47,57,45,55
3450 DATA 37,67,35,65,28,78,27,77
3460 DATA 44,54,26,76,38,68,17,87
3470 DATA 18,88,34,64,25,75,16,86
3480 DATA 48,24,74,15,85,14,84,43
3490 DATA 53,33,62,23,73,52,42,62
3500 DATA 32,83,13,72,22,12,82,41
3510 DATA 51,31,61,21,71,11,81,58
3520 CLS:RETURN
3530 REM*****************************
3540 FOR Z = 11 TO 88:Z(Z) = M(Z):NEXT Z
3550 FOR Z = 11 TO 88:X = Z - 10 * INT(Z/10)
3560 IF X = 0 OR X = 9 THEN 3580
3570 M(Z) = Z(Z + 9 - X * 2)
```

```
3580 NEXT Z
3590 FOR Z = 11 TO 88:M = M(Z)
3600 IF M >= A THEN M(Z) = M(Z) + PN - P
3610 IF M <= TN AND M >= AN THEN M(Z) = M(Z) - PN
+ P
3620 NEXT Z
3630 GOSUB 2580
3640 RETURN
3650 REM*******************************
3660 MC = INT(RND(-N) * 4) + 1
3670 ON MC GOSUB 3690,3700,3710,3720
3680 FOR J = 1 TO 1000:NEXT J:RETURN
3690 PRINT "BIEN HECHO!":RETURN
3700 PRINT "BUEN MOVIMIENTO":RETURN
3710 PRINT "BIEN MOVIDA, CAMPEON!":RETURN
3720 PRINT "ME DEJAS ASOMBRADA...":RETURN
3730 LPRINT "*************************************"
3740 GOSUB 3800
3750 FOR X = 8 TO 1 STEP -1
3760 LPRINT TAB(5);X;" ";
3770 FOR Y = 10 TO 80 STEP 10
3780 LPRINT CHR$(M(X + Y));" ";
3790 NEXT Y:LPRINT X:NEXT X
3800 LPRINT:LPRINT TAB(9);"A B C D E F G H":LPRIN

3810 RETURN
```

Si quieres que el ordenador juegue consigo mismo, cambia la línea 2580 de la siguiente manera:

2580 CLS:PRINT ''Jugador''PR''en la parte alta''

y cambia las primeras diez líneas del programa de esta otra manera:

```
15 PR = 1
20 GOSUB 2970
30 GOTO 60
40 GOSUB 2580
50 A$ = ''X''
60 IF PR = 2 THEN PR = 1: GOTO 80
```

```
70 PR = 2
80 IF A$ = "S" THEN END
90 IF A$ = "X" THEN PRINT "INTERCAMBIAMOS LAS
   PIEZAS": GOSUB 3540:A$ = ""
100 IF PR = 2 THEN GOSUB 3730
```

Si deseas que el programa te vaya dando por la impresora información acerca del desarrollo del juego, añade las siguientes líneas:

```
3730 IF RND >.08 THEN RETURN
3735 LPRINT "*******************************"
3740 GOSUB 3800
```

COMO FUNCIONA EL PROGRAMA

Vamos a echar ahora un vistazo a algunas de las partes de este programa con el fin de que el lector sepa con cuáles se ha de meter en primer lugar en el caso de que se decida a modificar el programa para mejorar su juego.

Como ya explicaba en la introducción a este capítulo, el programa empieza con una serie de llamadas de subrutina que se irán repitiendo cíclicamente según avanza el programa. Después de la primera sentencia REM que identifica el programa, el ordenador pasa a la subrutina, que empieza en la línea 2970, que inicializa las variables. Posteriormente, la línea 30 salta a la 60 que llama a la subrutina de impresión del tablero en la línea 2580. Esta rutina, como el resto de las rutinas de impresión del tablero de esta parte, empieza con una orden de borrar la pantalla (CLS). Si se dispone de una clave HOME o similar (como PRINT@,0,PRINT AT 0, o LOCATE 1,1), se ha de poner ésta en lugar de la CLS. Con esto el tablero se imprimirá simplemente sobre su antigua imagen lo que da la impresión, efectista por demás, de que las fichas se mueven. Esto es mucho más satisfactorio que borrar la pantalla cada vez que haya que imprimir el tablero.

Después de imprimir el tablero, el programa continúa hasta pasar la línea 60 con la sección de jugadas del ordenador. En primer lugar, éste comprueba la situación de A$, que es una serie que fija el jugador humano después de cada jugada. Si A$ es igual que "T", en ese caso el jugador humano ha señalado un deseo de abandonar la partida y el

programa finaliza. Si A$ ha sido ajustado en "I" (para intercambio) el programa continúa con la subrutina que empieza en la línea 3540 para intercambiar lados. Si A$ es igual que "C", el ordenador avanza siguiendo la subrutina que comienza en la línea 3730 para sacarnos una copia del tablero por la impresora.

Es ahora cuando empieza en serio el juego del ordenador. En primer lugar, la matriz F (que será la que contenga la posición de las fichas del ordenador) se completa con ceros y la variable que cuenta las fichas (U) se ajusta a cero. El bucle de las líneas 150 y 160 atraviesa cada casilla del tablero contando las fichas (y cada vez que encuentra una, U va aumentando y almacenando su localización en los elementos de la matriz F). Cuando encuentra el rey (variable RN, de «rey negro») se ajusta igual a éste la variable SR (de «señalador rey») de modo que el ordenador sabe en todo momento dónde está su rey. Si U no pasa de tres (línea 160), el ordenador irá hasta la línea 2230 y se dará por vencido. Este no cuenta con una estrategia de fin de partida y tiende a enredarse si no tiene suficientes fichas.

La línea 170 hace saltar la acción hasta la 650 donde una larga rutina verifica si le pueden dar jaque al rey. En esta parte del programa cada sección del código comprueba el peligro que supone en particular cada ficha del contrincante. La primera parte, desde la línea 650 a la 710 verifica el peligro por parte de los caballos del jugador humano, mientras las siguientes (730 a 890) vigilan a la reina, los alfiles, las torres y los peones. Si no encuentra ningún peligro, de modo que la corriente del programa no es reencauzada por los numerosos GOTOS de esta sección, la línea 890 retrotrae el programa a la 180 en donde empieza a buscar jugada.

Las líneas 180 y 190 reordenan los contenidos de la matriz F de modo que el rey sea la última ficha de esta matriz. En general, el ordenador intentará mover sus fichas en términos de su posición en esta matriz, así pues, como el rey está al final, sólo lo moverá si considera que no hay otra jugada que merezca la pena. Como dije con antelación al listado, esta es una de aquellas ideas simplemente aproximativas, pero ingeniosas, con respecto al ajedrez y está basada en el principio de que cuanto menos se mueva al rey, menos ocasiones de peligro correrá éste. Por supuesto, esto no siempre es cierto. No obstante, para el tipo de ajedrez que juega el programa no es del todo irracional el partir de este principio y desde luego, es menos peligroso que la alternativa de tener al rey alrededor del tablero exponiéndose a cada jugada.

Seguidamente se ajusta D en 0, 1, ó 2 hacia la línea 210 y se vuelve a ajustar en 0 en la línea 210 si el jugador humano ha señalado "J" para «jaque». D determina en qué punto de la matriz F empezará a buscar su jugada el ordenador. Se introduce aquí el elemento aleatorio de forma que el ordenador no siempre hará lo mismo si se le confronta a dos situaciones de tablero similares. Se añade uno a D si D es menor que U (es decir, menor que el número total de fichas que el ordenador tiene sobre el tablero) y después Z se ajusta igual que ese elemento de la matriz T que es la posición de la ficha del ordenador en el tablero.

Tras esto, se llama a la subrutina que comienza en la línea 280. Si se fija en esta subrutina, se dará cuenta de que las líneas 280 a 320 determinan de qué ficha se está ocupando. El ordenador juega con las negras y para facilitar la tarea de seguir lo que está sucediendo en el listado se utiliza DN para indicar la reina («dama negra»), TN es la torre, AN el alfil, CN el caballo y PN el peón. Así, el ordenador en esta pequeña subrutina lo único que hace es descubrir qué ficha tiene entre manos dirigiéndose posteriormente a una subrutina más importante que es la de comprobar si esta ficha puede comer alguna de su contrincante.

A estas alturas, seguro que el lector ya se ha hecho una idea de cómo funciona el programa. Lo que esencialmente hace es localizar sus fichas. Si no le pueden dar jaque, pone las fichas en el orden que le dicta la disposición del tablero (esto lo explicaré cuando lleguemos al final del listado del programa) con el rey al final de esta lista y después va comprobando ficha por ficha el modo de comer a su contrincante.

La variable más importante (después de SR, el señalizador del rey) es MM (abreviación de «movimiento de la máquina»). Encontrará MM en la línea 240. MM se pone inicialmente a cero y sólo se ajusta a uno cuando el ordenador ha encontrado la jugada pasando así a la subrutina que comienza en la línea 2500. Aquí comprueba que: *a)* la jugada pensada no consiste en mover su rey sin una causa justificada; *b)* no va a mover un peón tontamente, y *c)* que no está a punto de comer al rey del jugador humano. Cualquiera de estos tres casos provocarían que el señalizador MM tuviera que ser ajustado de nuevo en cero volviendo en ese caso la acción a la línea 230 para buscar otra jugada (si el ordenador considera que se da el caso *c)*, imprime JAQUE y continúa pensando su jugada).

Si considera que no se encuentra en ninguno de estos tres casos, hace la jugada (la línea 2530 realiza los cambios necesarios en los

elementos de la matriz F, la matriz que de hecho contiene el tablero) y luego le indica al contrincante (usando las líneas 2540 y 2550) la jugada que va a hacer volviendo posteriormente al inicio del programa en donde se vuelve a imprimir el tablero.

Sin embargo, si MM no era igual a uno cuando el ordenador llegó a la línea 240, el ordenador seguirá avanzando hasta la 250, en donde la máquina comprueba si todavía le quedan fichas (de no quedarle, D sería igual a U) y si verifica que le quedan retrocede a la línea 220, en donde se aumenta uno a D y el proceso vuelve a empezar. Puede suceder, no obstante, que D sea igual a U, en cuyo caso el ordenador decide que no puede hacer una jugada correcta yendo entonces a la línea 2360 en donde busca una jugada que no implique comer a su contrincante. La siguiente sección, como ya hemos comentado, envía la acción a varias partes del programa, cada una de las cuales controlan fichas diferentes.

Si observamos la sección que comienza en la línea 2360, veremos que se inicia fijando D a un número aleatorio entre cero y cuatro. El uso de un RND doble asegura que tenderá a escoger, dentro de esa gama, números bajos mucho más a menudo que altos. Z se ajusta igual al elemento D° de la matriz F (como antes, cuando lo que buscaba era un modo de comer) y luego las líneas que van de la 2390 a la 2440 envían el programa a las secciones de código que controlan sus movimientos.

No nos detendremos en todas y cada una de las secciones que controlan los movimientos de las diferentes fichas porque todas ellas funcionan de un modo similar. Comprendiendo el principio básico de una se puede deducir fácilmente cómo funcionan las otras.

Observaremos la sección que se encarga del juego del caballo, puesto que en la introducción a este capítulo ya hemos hecho algún comentario sobre cómo se mueven los caballos. Fíjese el lector en la sección del código que empieza en la línea 1790. Como se dará cuenta cuando nos detengamos en la sección de inicialización del programa, se establece una matriz para cada ficha, matriz que contiene las posibles jugadas de esa ficha (en términos de un desplazamiento numérico desde la casilla inicial). La matriz D contiene las posibles jugadas de la reina, la A las que puede hacer el alfil, etc. Así pues, sabemos que C, al que se refiere la línea 1800, contiene las jugadas del caballo.

En esta línea la variable X se ajusta igual a Z (la casilla que ocupa la ficha) más un elemento (escogido aleatoriamente) de la matriz C. La línea 1810 verifica que el posible destino de la ficha no se encuen-

tra fuera del tablero (lo que sería el caso si su valor fuera inferior a 11 y superior a 88) o igual a 42 (lo que también querría decir que se encontraría fuera del tablero). Y es la variable que cuenta el número de jugadas potenciales intentadas. Se le aumenta uno en la línea 1830 y luego el ordenador comprueba el valor que contiene en ese momento M(X), que es la casilla de destino. Si encuentra que contiene el valor V (de «vacío») sabe que puede mover a esa casilla. Pero no salta a lo loco hasta M(X) sólo porque es una casilla que está vacía.

En primer lugar va a la subrutina que empieza en la línea 350 y comprueba si tal casilla se encuentra atacada por el contrincante. Al volver de ese destino comprueba (línea 1850) si MM es igual a uno. Si lo es, el ordenador sabe que la jugada que estaba considerando ha sido aprobada y con eso avanza hasta la sección de código que se encarga de hacer la jugada. Si Y es mayor de 20, lo que significa que se han intentado 20 jugadas sin resultado, en ese caso también dispara una vuelta de modo que vuelve a probar con una nueva ficha. Si encuentra que no se dan ninguna de estas dos condiciones, el ordenador regresa a la línea 1800 para conseguir una nueva jugada aleatoria del caballo.

Así es, en esencia, cómo se generan y comprueban todas y cada una de las posibles jugadas. El caso del caballo es un poco más sencillo, porque puede saltar sobre otras fichas. En los otros casos, el ordenador ha de comprobar que existe un paso abierto para que pueda realizarse la jugada que se pretende, pero esto es fácil de conseguir.

Avanzando ahora hasta la sección del programa que empieza en la línea 2970, se dará cuenta el lector de que es en este punto donde tiene lugar la inicialización. En primer lugar el ordenador indica al jugador humano: "PONGA EL TECLADO EN MAYUSCULAS Y A CONTINUACION PULSE RETURN". El ordenador calcula el tiempo que lleva esta operación (usando N como contador) y usa este valor para inicializar el generador de números aleatorios. Como en todos los programas de este libro, en el caso de tratarse de un ordenador en el que la función RANDOMIZE no se lleva a cabo del mismo modo, el lector ha de cambiar la línea para adecuarla al dialecto BASIC que utilice. La línea 3000 asegura que todas las variables serán tratadas como variables de números enteros. Se ha hecho así para que la velocidad de ejecución sea la máxima, pero puede excluirse, sin que el programa quede alterado por ello, si el ordenador no dispone de DEFINT. MM (recuerde que es «movimiento de la máquina») se

ajusta en cero en la línea 3010 y A$ (que contiene los mensajes del jugador humano, tales como el deseo de cambiar de lado) se iguala a una cadena nula.

Posteriormente se dimensionan un número determinado de matrices. M contiene el propio tablero, T las jugadas de la torre, A las del alfil, C las del caballo, D las de la reina, Z el mecanismo de cambio de lados, S la secuencia de prioridad que determina en qué orden se comprueban las casillas en busca de fichas, y F la localización de las fichas en el tablero antes de cada jugada.

Las dos líneas que siguen inicializan, como se señala a continuación, las variables, que son las fichas del jugador humano y del ordenador:

P – peón humano	PN – peón del ordenador
T – torre humana	TN – torre del ordenador
C – caballo humano	CN – caballo del ordenador
A – alfil humano	AN – alfil del ordenador
D – reina humana	DN – reina del ordenador
R – rey humano	RN – rey del ordenador
V – casilla vacía	

Las dos líneas siguientes (3050) completan la matriz M (que contendrá el tablero) con un valor ficticio (−99). Esto asegura que toda casilla a la que no se le dé posteriormente un valor será ignorada a la hora de ejecutar el programa. Este programa usa un tablero numerado de modo similar al que se ha mostrado al inicio de este capítulo, pero éste está girado de modo que el lateral derecho de aquél ha pasado a ser la parte superior de éste y el lateral izquierdo, la parte inferior. Quedando la siguiente disposición:

```
18  28  38  48  58  68  78  88  *8
17  27  37  47  57  67  77  87  *7
16  26  36  46  56  66  76  86  *6
15  25  35  45  55  65  75  85  *5
14  24  34  44  54  64  74  84  *4
13  23  33  43  53  63  73  83  *3
12  22  32  42  52  62  72  82  *2
11  21  31  41  51  61  71  81  *1
*******************************
A   B   C   D   E   F   G   H
```

La sección siguiente del programa completa las matrices con un bucle Z (en la línea 3070) poniendo todas las fichas en su lugar para el inicio del programa. La siguiente sección es claramente obvia cuando se examina, con la línea 3260 que da los valores a la matriz C para las jugadas del caballo, la 3280 que hace lo mismo con las de la torre y la 3340 con las del alfil. El indicador de DATA vuelve entonces a la línea 3290, el inicio de la información de la torre, de modo que la matriz D pueda completarse con las localizaciones tanto de la torre como del alfil, ya que la reina puede jugar por ambas fichas.

La línea 3430 completa la matriz S con los números que aparecen en las sentencias DATA de las líneas 3430 a 3510. Quizá sean éstas las principales causantes de que el programa juegue tan rápido como lo hace a la hora de ejecutarlo. Estas series de números constituyen el orden en que se comprueban las casillas. Si compara esta secuencia con la del «tablero numerado» no le será difícil calcular la secuencia. Comienza con las casillas situadas enfrente de la posición inicial de los dos peones clave seguida de las casillas situadas a ambos lados de éstos, antes de retroceder a comprobar las casillas en las que los peones clave comienzan el juego. Una vez que se haya abierto paso a través de esta lista, es probable que quiera modificar la secuencia. Esta matriz, más que ninguna otra del programa, determina el tipo de juego que jugará el ordenador. Si el lector no está de acuerdo con la secuencia que yo he establecido, cámbiela sin dudarlo un momento y observe el efecto que esto produce en el juego del ordenador.

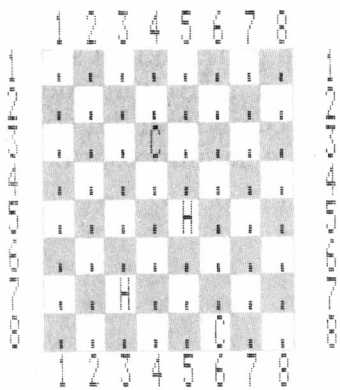
La rutina de «cambio de lados» va de la línea 3540 a la 3640. Cuando el ordenador pierde una ficha se generan una serie de comentarios (del tipo de "Bien movida campeón", "Bien hecho") en la rutina que va de la línea 3660 a 3720 y la sección final del código vierte la situación del tablero en ese momento a la impresora.

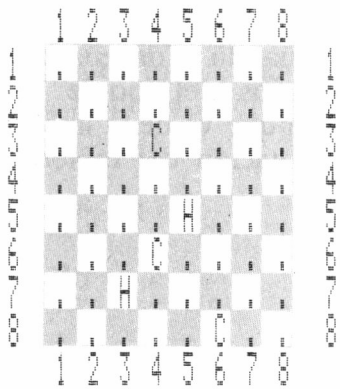
GOMOKU

El GOMOKU es un juego que se aprende con mucha facilidad, pero en el que es casi imposible ganar. El ordenador juega extremadamente bien en este programa, que está basado en uno escrito por Graham Charlton.

El juego consiste en intentar colocar cinco de las fichas de uno (las H) en fila en cualquier dirección antes de que lo consiga el ordenador.

Eche un vistazo de las posiciones de tablero que aquí se muestran y que están sacadas de una partida que yo jugué contra el programa. El estudio de éstas le mostrará cómo se desarrolla el juego y el modo de jugarlo.





POR FAVOR, CUAL ES TU MOVIMIENTO.
?

POR FAVOR, CUAL ES TU MOVIMIENTO...
?

```
  1 2 3 4 5 6 7 8
1 . . . . . . . . 1
2 . . . . . . . . 2
3 . . . C . . . . 3
4 . . . . . . . . 4
5 . . . . H . . . 5
6 . . . C C . . . 6
7 . . H . H . . . 7
8 . . . . . C . . 8
  1 2 3 4 5 6 7 8
```

POR FAVOR, CUAL ES TU MOVIMIENTO...
?

```
  1 2 3 4 5 6 7 8
1 . . . . . . . . 1
2 . . . . . . . . 2
3 . . . C . . . . 3
4 . . . . . . . . 4
5 . . . . H C H . 5
6 . . . C C . . . 6
7 . . H . H . . . 7
8 . . . . . C . . 8
  1 2 3 4 5 6 7 8
```

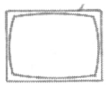
POR FAVOR, CUAL ES TU MOVIMIENTO...
?

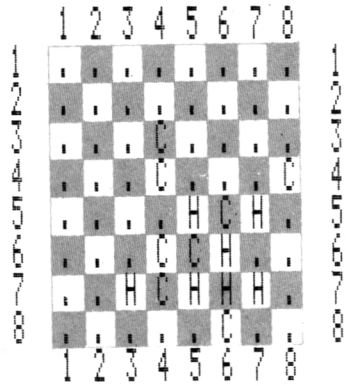
```
  1 2 3 4 5 6 7 8
1 . . . . . . . . 1
2 . . . . . . . . 2
3 . . . C . . . . 3
4 . . . . . . . C 4
5 . . . . H C H . 5
6 . . . C C H . . 6
7 . . H . H . H . 7
8 . . . . . C . . 8
  1 2 3 4 5 6 7 8
```

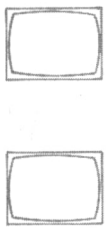
POR FAVOR, CUAL ES TU MOVIMIENTO...
?

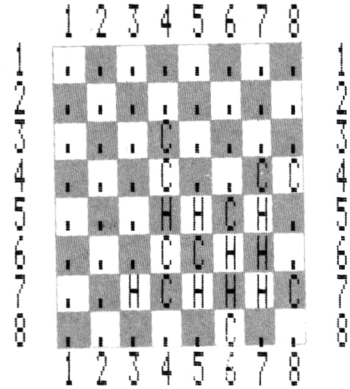
```
  1 2 3 4 5 6 7 8
1 . . . . . . . . 1
2 . . . . . . . . 2
3 . . . C . . . . 3
4 . . . C . . . C 4
5 . . . . H C H . 5
6 . . . C C H . . 6
7 . . H . H . H . 7
8 . . . . . C . . 8
  1 2 3 4 5 6 7 8
```

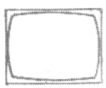
POR FAVOR, CUAL ES TU MOVIMIENTO...
?











POR FAVOR, CUAL ES TU MOVIMIENTO..POR FAVOR, CUAL ES TU MOVIMIENTO...
? ?

Y este es el listado del GOMOKU:

```
 10 REM Gomoku
 20 GOSUB 750
 30 GOSUB 130
 40 GOSUB 240
 50 GOSUB 130
 60 GOSUB 320
 70 GOSUB 130
 80 IF L > 3 THEN PRINT:PRINT "YO GANO!!":END
 90 GOTO 40
100 E = A
110 E = E + N:IF A(E) <> Z THEN RETURN
120 K = K + 1:GOTO 110
130 CLS
140 PRINT:PRINT:PRINT
150 PRINT TAB(12);"1 2 3 4 5 6 7 8"
160 FOR A = 1 TO 8:PRINT TAB (8);A;" ";
170 FOR B = 2 TO 9
180 PRINT CHR$(A(A*10 + B));" ";
190 NEXT B
200 PRINT A
210 NEXT A
```

```
220 PRINT TAB(12);"1 2 3 4 5 6 7 8"
230 RETURN
240 PRINT:PRINT
250 PRINT "POR FAVOR, CUAL ES TU MOVIMIENTO..."
260 INPUT G
270 G = G + 1
280 IF G < 12 OR G > 89 OR A(G) <> 46 THEN 260
290 Z = H
300 A(G) = Z
310 RETURN
320 A = G
330 L = 0
340 FOR X = 1 TO 4:K = 0:N = X(X)
350 GOSUB 100
360 N = -N:GOSUB 100
370 IF K > L THEN L = K
380 NEXT X
390 IF L > 3 THEN PRINT:PRINT "TU GANAS!!":END
400 T = 1
410 IF T <> 2 THEN Z = C
420 IF T = 2 THEN Z = H
430 G = 0:H1 = 0:L = 0
440 FOR A = 12 TO 89
450 M = 0
460 IF A(A) <> 46 THEN 570
470 FOR X = 1 TO 4:K = 0:N = X(X)
480 GOSUB 100
490 N = -N:GOSUB 100
500 IF K > L THEN H1 = 0:L = K
510 IF L <> K THEN 540
520 IF T = 1 AND L < 4 OR (T = 2 OR T = 3) AND L
< 2 THEN 540
530 M = M + 1
540 NEXT X
550 IF M <= H1 THEN 570
560 H1 = M:G = A
570 NEXT A
580 IF H1 <> 0 THEN 650
590 T = T + 1:IF T <> 4 THEN 410
600 A = 1
610 G = INT(RND(-1) * 77) + 13
620 IF A(G) = 46 THEN 650
630 A = A + 1:IF A < 100 THEN 610
```

```
640 PRINT:PRINT "ME RINDO ":PRINT "ANTE UN GRAN
MAESTRO!!":END
650 A(G) = C
660 Z = C:A = G:L = 0
670 FOR X = 1 TO 4
680 K = 0
690 N = X(X)
700 GOSUB 100
710 N = -N:GOSUB 100
720 IF K > LTHEN L = K
730 NEXT X
740 RETURN
750 CLS
760 DIM A(100),X(4)
770 FOR C = 1 TO 8
780 FOR B = 2 TO 9
790 A(C * 10 + B) = 46
800 NEXT B
810 NEXT C
820 FOR Q = 1 TO 4
830 READ Z:X(Q) = Z
840 NEXT Q
850 DATA 1,9,10,11
860 H = ASC("H"):C = ASC("C")
870 PRINT:PRINT "Oprima S si desea empezar"
880 PRINT "o bien N si no es asi"
890 N = 0
900 N = N + 1
910 A$ = INKEY$
920 IF A$ <> "s" AND A$ <> "S" AND A$ <> "n" AND
A$ <> "N" THEN 900
930 RANDOMIZE
940 CLS
950 IF A$ = "s" OR A$ = "S" THEN RETURN
960 FOR J = 1 TO INT(RND(-1) * 12) + 1
970 READ Z
980 NEXT J
990 A(Z) = C
1000 RETURN
1010 DATA 34,35,44,46,47,54,55,56,57,66
```

CUATRO EN RAYA

Como su propio nombre indica, el objetivo del CUATRO EN RAYA es conseguir colocar cuatro de las fichas de uno (las H) en fila en cualquier dirección antes de que el ordenador lo consiga con las suyas (las C).

El jugador humano indica la jugada que va a hacer especificando la columna a la que quiere mover su ficha. Tras esto la ficha se sitúa en la posición más baja disponible dentro de esa columna.

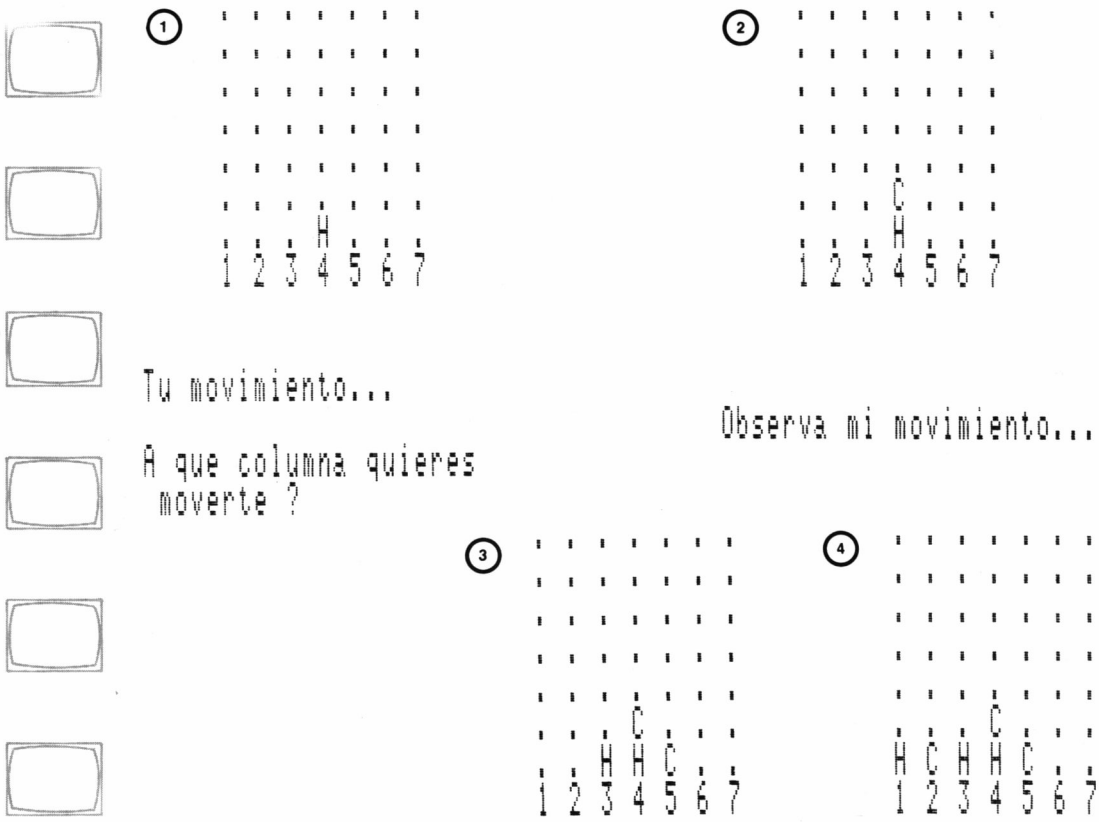
El ordenador juega bastante bien y es sorprendentemente rápido, si se considera el número de veces que puede llegar a atravesar esos bucles dentro del programa.

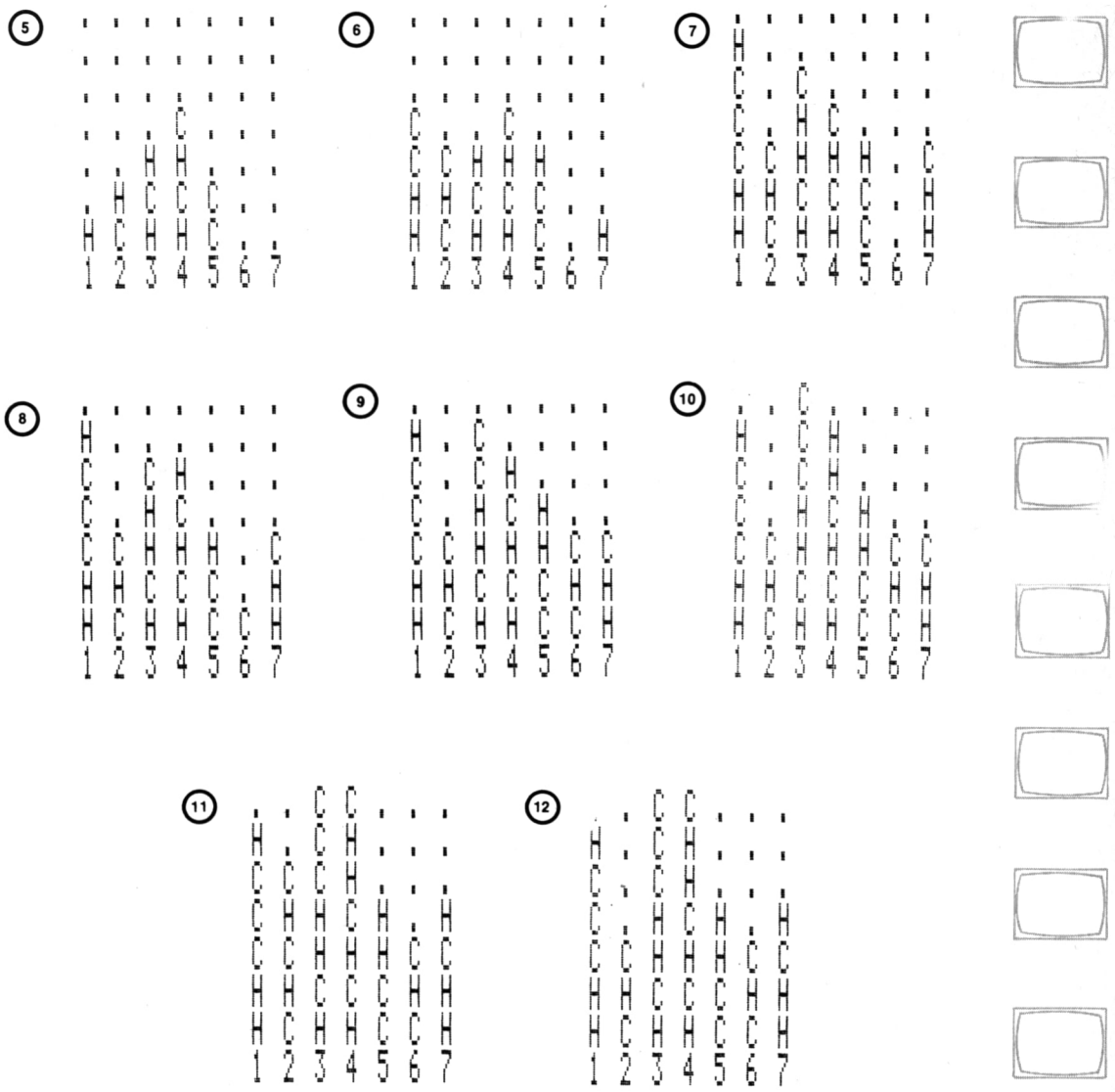
La primera versión escrita a mano de este programa no me dejó particularmente satisfecho, porque me daba la impresión de que había utilizado la «fuerza bruta» como modo de solucionar los problemas que se planteaban. Estaba seguro de que tenía que existir un modo mucho más inteligente de hacerlo. Sin embargo, continué con él y lo introduje en el ordenador. El programa ganó la primera partida y, aun cuando jugó de un modo un tanto extraño, supe que me encontraba ante un ganador, independientemente del modo en que había enfocado la programación. Posteriormente, cuando volví a meterme con él me di cuenta de que si el programa estaba correctamente estructurado (como de hecho lo estaba), si su código no era redundante (y por lo que se veía no lo era) y se ejecutaba con rapidez y facilidad, no había por qué andar «jugueteando» con él para hacerlo más complicado. Un código transparente es siempre mejor que un código inteligente, pero excesivamente complicado, el cual, aunque ocupe menos espacio y se ejecute unos cuantos microsegundos más rápido, es casi imposible de modificar.

Así pues, el programa que presento aquí es mi versión «por la fuerza bruta». Me he mostrado generoso a la hora de abastecerlo con sentencias REM de modo que el lector encontrara poca o ninguna dificultad a la hora de calcular qué es lo que hace cada sección del código. Asimismo, tras haber jugado unas cuantas partidas con el programa en su forma presente, al lector no debería resultarle demasiado difícil modificarlo.

En esta versión del CUATRO EN RAYA, el ordenador siempre permite que el jugador humano haga la primera jugada y basa su jugada inicial en la jugada hecha por el jugador humano. Es posible que el lector desee modificar el programa de modo que se dé también la posibilidad de que el ordenador abra el juego.

Antes de pasar al listado, observe esta partida jugada por mí mismo contra el ordenador.





Aquí tenemos el listado de CUATRO EN RAYA:

```
10 REM Cuatro en raya
20 GOSUB 1090:REM INICIALIZACION
30 GOSUB  860:REM IMPRIME TABLERO
40 GOSUB  680:REM COMPRUEBA VICTORIA
```

```
 50 GOSUB  980:REM MOVIMIENTO DEL HUMANO
 60 GOSUB  860:REM IMPRIME TABLERO
 70 GOSUB  680:REM COMPRUEBA VICTORIA
 80 GOSUB  110:REM MOVIMIENTO DE LA COMPUTADORA
 90 GOTO 30
100 REM ****************************
110 REM MOVIMIENTO DE LA COMPUTADORA
120 PRINT:PRINT "Observa mi movimiento..."
130 B = 10
140 B = B + 1
150 IF A(B) = -9 THEN 180
160 IF A(B) = C THEN X = C:GOTO 210
170 IF A(B) = H THEN X = H:GOTO 210
180 IF B < 77 THEN 140
190 GOTO 480
200 REM ****************************
210 REM PELIGRO O POSIBILIDAD DE CUATRO EN RAYA
?
220 REM EN HORIZONTAL
230 IF A(B + 1) = X AND A(B + 2) = X AND A(B + 3
) = V AND A(B + 13) <> V THEN MOVIMIENTO = B + 3:G
OTO 650
240 IF A(B - 1) = X AND A(B - 2) = X AND A(B - 3
) = V AND A(B + 7) <> V THEN MOVIMIENTO = B - 3:GO
TO 650
250 IF A(B + 1) = X AND A(B + 2) = X AND A(B - 1
) = V AND A(B + 9) <> V THEN MOVIMIENTO = B - 1:GO
TO 650
260 IF A(B - 1) = X AND A(B + 2) = X AND A(B + 1
) = V AND A(B + 11) <> V THEN MOVIMIENTO = B + 1:G
OTO 650
270 IF A(B + 1) = X AND A(B - 1) = X AND A(B + 2
) = V AND A(B + 12) <> V THEN MOVIMIENTO = B + 2:G
OTO 650
280 IF A(B + 1) = X AND A(B - 1) = X AND A(B - 2
) = V AND A(B + 8) <> V THEN MOVIMIENTO = B - 2:GO
TO 650
290 IF A(B - 1) = X AND A(B - 2) = X AND A(B + 1
) = V AND A(B + 11) <> V THEN MOVIMIENTO = B + 1:G
OTO 650
300 REM VERTICAL
310 IF B > 20 THEN IF A(B - 10) = X AND A(B - 20
) = X AND A(B + 10) = V AND A(B + 20) <> V THEN MO
VIMIENTO = B + 10:GOTO 650
```

```
320 REM DIAGONALES
330 IF A(B + 11) = X AND A(B + 22) = X AND A(B -
 11) = V AND A(B - 1) <> V THEN MOVIMIENTO = B - 1
1:GOTO 650
340 IF A(B + 9) = X AND A(B + 18) = X AND A(B -
9) = V AND A(B + 1) <> V THEN MOVIMIENTO = B - 9:G
OTO 650
350 REM ******************************
360 REM HACER/BLOQUEAR UNO DE TRES?
370 REM EN HORIZONTAL
380 IF A(B + 1) = X AND A(B + 2) = V AND A(B + 1
2) <> V THEN MOVIMIENTO = B + 2:GOTO 650
390 IF A(B + 1) = X AND A(B - 1) = V AND A(B + 9
) <> V THEN MOVIMIENTO = B - 1:GOTO 650
400 IF A(B - 1) = X AND A(B - 2) = V AND A(B + 8
) <> V THEN MOVIMIENTO = B - 2:GOTO 650
410 REM EN VERTICAL
420 IF A(B + 10) = X AND A(B - 10) = V AND A(B)
<> V THEN MOVIMIENTO = B - 10:GOTO 650
430 REM EN DIAGONAL
440 IF A(B + 9) = X AND A(B - 9) = V AND A(B + 1
) <> V THEN MOVIMIENTO =B - 9:GOTO650
450 IF B > 11 THEN IF A(B + 11) = X AND A(B - 11
) = V AND A(B - 1) <> V THEN MOVIMIENTO = B - 11:G
OTO 650
460 GOTO 180
470 REM ******************************
480 REM MOVIMIENTOS INDIVIDUALES
490 FOR N = 1 TO 3
500 M(N) = 0
510 NEXT N
520 CUENTA = 0
530 FOR B = 11 TO 77
540 IF A(B)<>C AND A(B)<>H THEN 600
550 IF A(B + 1) = V AND A(B + 11) <> V THEN CUEN
TA = CUENTA + 1:M(CUENTA) = B + 1
560 IF A(B - 1) = V AND A(B + 9) <> V THEN CUENT
A = CUENTA + 1:M(CUENTA) = B - 1
570 IFA(B-10)=V AND A(B)<>V THEN CUENTA=CUENTA+1
:M(CUENTA)=B-10
580 IF A(B - 11) = V AND A(B - 1) <> V THEN CUEN
TA = CUENTA + 1:M(CUENTA) = B - 11
590 IF A(B - 9) = V AND A(B + 1) <> V THEN CUENT
A = CUENTA + 1:M(CUENTA) = B - 9
```

```
600 NEXT B
610 IF CUENTA <> 0 THEN 640
620 PRINT:PRINT "Creo que deberiamos llamarlo un
'ahogado'"
630 PRINT:PRINT:PRINT:END
640 MOVIMIENTO = M(INT(RND(-1) * CUENTA) + 1)
650 A(MOVIMIENTO) = C
660 RETURN
670 REM ****************************
680 REM COMPRUEBA POSIBLE JUGADA GANADORA
690 X = H
700 B = 10
710 B = B + 1
720 IF A(B) <> X THEN 770
730 IF A(B + 1) = X AND A(B + 2) = X AND A(B + 3
) = X THEN 800
740 IF B > 30 THEN IF A(B - 10) = X AND A(B - 20
) = X AND A(B - 30) = X THEN 800
750 IF B > 33 THEN IF A(B - 11) = X AND A(B - 22
) = X AND A(B - 33) = X THEN 800
760 IF B > 27 THEN IF A(B - 9) = X AND A(B - 18)
 = X AND A(B - 27) = X THEN 800
770 IF B < 77 THEN 710
780 IF X = H THEN X = C:GOTO 700
790 RETURN
800 REM ENCONTRADOS CUATRO EN RAYA
810 PRINT:PRINT
820 IF X = H THEN PRINT "Me has vencido, humano.
.."
830 IF X=C THEN PRINT "Te he vencido, humano..."
840 PRINT:PRINT:PRINT:END
850 REM ****************************
860 REM IMPRIME EL TABLERO DE JUEGO
870 CLS:PRINT:PRINT
880 FOR K = 10 TO 70 STEP 10
890 PRINT TAB(5);
900 FOR J = 1 TO 7
910 PRINT CHR$(A(K + J));" ";
920 NEXT J
930 NEXT K
940 PRINT TAB(5);"1 2 3 4 5 6 7"
950 PRINT:PRINT
960 RETURN
```

```
 970 REM ***************************
 980 REM MOVIMIENTO DEL HUMANO
 990 PRINT "Tu movimiento...":PRINT
1000 PRINT "A que columna quieres"
1010 INPUT " moverte ";J
1020 Z = J
1030 Z = Z + 10
1040 IF A(Z + 10) = V THEN 1030
1050 IF A(Z) = V THEN A(Z) = H:RETURN
1060 PRINT "No puedes moverte ahi"
1070 GOTO 1000
1080 REM ***************************
1090 REM INICIALIZACION
1100 CLS
1110 DEFINT A-Z
1120 RANDOMIZE
1130 DIM A(109),M(30),P(6)
1140 V = ASC(".")
1150 H = ASC("H"):C=ASC("C")
1160 FOR B = 1 TO 109
1170 A(B) = V
1180 D = B - 10 * INT(B / 10)
1190 IF D = 0 OR D > 7 OR B < 11 OR B > 77 THEN A
(B) = -9
1200 NEXT B
1210 RETURN
```

DAMAS

El juego de las DAMAS cuenta con una larga y honorable historia. R. C. Bell (en su libro *Discovering Old Board Games,* Shire Publications, Aylesbury, UK, 1980), dice que se inventó en torno al 1100 «probablemente en el sur de Francia utilizando las fichas del "backgammon" en un tablero de ajedrez y el método Alquerque para comer las fichas». La *Encyclopedia of Sports, Games and Pastimes* (Fleetway House, London, 1935) retrotrae sus orígenes todavía más lejos en el tiempo: «Formas de este juego se conocían ya en el Antiguo Egipto, Grecia y Roma, al tiempo que se jugaba mucho más a mediados del siglo XVII de lo que se juega hoy día.»

Independientemente de su edad, es un juego muy popular en todo el mundo y muchos países europeos tienen sus propias variedades regionales. En general, en la Europa continental, por ejemplo, se suele jugar con un tablero de 100 casillas y cada jugador empieza la partida con veinte fichas. Esta modalidad se desarrolló a principios del siglo XVIII.

Este programa de damas juega la modalidad con la que quizá el lector esté más familiarizado. Juega con rapidez y razonablemente bien y, aunque carece de mecanismo para señalar el final del juego, a menudo puede llevar a un derrumbamiento dramático en el último momento.

En las damas cada contrincante, usted y el ordenador por turno, intenta ya sea comer las fichas del contrario (lo que se realiza saltando sobre ellas y eliminándolas posteriormente del tablero) o confinarlas de modo que a éste no le sea posible moverlas de donde están.

Generalmente se suele jugar en un tablero de 64 casillas en el que alternan las blancas y las negras. En este programa el tablero (como

puede comprobarlo en la muestra que le ofrecemos a continuación) consiste en una serie de puntos en el que las fichas del jugador humano están marchadas con H (de humano) y las del ordenador con C (de «computadora competente»).

Al principio del juego, el jugador humano tiene todas sus fichas en la parte inferior de la pantalla y va jugando hacia arriba y el ordenador tiene las suyas en la parte superior y va jugando hacia abajo. Tal como está establecido en el programa, el ordenador hace siempre la primera jugada. Si el lector prefiere que sea el jugador humano quien abra el juego, no tiene más que tachar la línea 50.

Todas las jugadas se hacen en diagonal y hacia adelante, es decir, hacia el lado de partida del contrincante. Las fichas se convierten en «damas» cuando llegan al otro lado del tablero, el opuesto de donde han partido. Las damas del jugador humano aparecen como K y las del ordenador con el símbolo del dólar. Las damas se pueden mover hacia adelante y hacia atrás. Después de comer una ficha, saltando sobre ella a la casilla vacía situada inmediatamente después, se puede mover de nuevo siempre que sea posible volver a comer con la misma ficha.

He aquí una serie de ejecuciones del programa:

```
COMPUTADORA: 0
     HUMANO: 0

  1 2 3 4 5 6 7 8
8 . C . C . C . C 8
7 C . C . C . C . 7
6 . C .   . C . C 6
5   .   . C .   . 5
4 .   .   .   .   4
3 H . H . H . H . 3
2 . H . H . H . H 2
1 H . H . H . H . 1
  1 2 3 4 5 6 7 8
```

```
COMPUTADORA: 0
     HUMANO: 0

  1 2 3 4 5 6 7 8
8 . C .   . C . C 8
7 C . C . C . C . 7
6 .   . C . C . C 6
5 C .   . C .   . 5
4 . H .   .   . H 4
3   . H . H . H . 3
2 . H . H .   . H 2
1 H . H . H . H . 1
  1 2 3 4 5 6 7 8
```

```
COMPUTADORA: 0
     HUMANO: 1

   1 2 3 4 5 6 7 8
8  . ▒ . ▒ . C . C  8
7  C . C . C . C .  7
6  . C . H . C . C  6
5  C . ▒ . C . ▒ .  5
4  . ▒ . ▒ . ▒ . H  4
3  H . H . H . H .  3
2  . H . H . H . H  2
1  ▒ . H . ▒ . H .  1
   1 2 3 4 5 6 7 8
```

```
COMPUTADORA: 0
     HUMANO: 2

   1 2 3 4 5 6 7 8
8  . K . ▒ . C . C  8
7  C . ▒ . C . C .  7
6  . ▒ . ▒ . C . C  6
5  C . C . C . ▒ .  5
4  . ▒ . ▒ . ▒ . H  4
3  H . H . H . H .  3
2  . H . H . H . H  2
1  ▒ . H . ▒ . H .  1
   1 2 3 4 5 6 7 8
```

```
COMPUTADORA: 2
     HUMANO: 2

   1 2 3 4 5 6 7 8
8  . ▒ . ▒ . C . C  8
7  C . ▒ . ▒ . C .  7
6  . K . C . C . ▒  6
5  ▒ . C . C . C .  5
4  . ▒ . ▒ . ▒ . H  4
3  H . ▒ . H . H .  3
2  . ▒ . H . H . H  2
1  $ . H . ▒ . H .  1
   1 2 3 4 5 6 7 8
```

```
COMPUTADORA: 2
     HUMANO: 3

   1 2 3 4 5 6 7 8
8  . ▒ . ▒ . C . C  8
7  C . ▒ . ▒ . C .  7
6  . ▒ . C . C . ▒  6
5  ▒ . ▒ . C . C .  5
4  . ▒ . K . ▒ . H  4
3  H . ▒ . H . H .  3
2  . ▒ . H . H . H  2
1  $ . H . ▒ . H .  1
   1 2 3 4 5 6 7 8
```

```
COMPUTADORA: 4
     HUMANO: 3

   1 2 3 4 5 6 7 8
8  . . . . . C . C  8
7  C . . . . . C .  7
6  . . . C . C . .  6
5  . . . . . . C .  5
4  . . . . . . . H  4
3  H . . . H . H .  3
2  . . . . . H . H  2
1  $ . H . C . H .  1
   1 2 3 4 5 6 7 8
```

```
COMPUTADORA: 4
     HUMANO: 3

   1 2 3 4 5 6 7 8
8  . . . . . . . C  8
7  . . . C . . . .  7
6  . C . C . C . C  6
5  H . . . . . C .  5
4  . . . H . . . H  4
3  . . . . . H . .  3
2  . . . . . H . H  2
1  $ . H . $ . H .  1
   1 2 3 4 5 6 7 8
```

```
COMPUTADORA: 4
     HUMANO: 4

   1 2 3 4 5 6 7 8
8  . . . . . . . .  8
7  . . H . C . C .  7
6  . . . C . C . C  6
5  . . . . . C . .  5
4  . . . H . . . H  4
3  . . . . . H . .  3
2  . . . . . H . H  2
1  $ . H . $ . H .  1
   1 2 3 4 5 6 7 8
```

```
COMPUTADORA: 6
     HUMANO: 6

   1 2 3 4 5 6 7 8
8  . . . . . . . .  8
7  . . . . C . C .  7
6  . . . . . . . C  6
5  . . H . . . C .  5
4  . $ . . . H . H  4
3  . . . . . . H .  3
2  . . . . . C . H  2
1  . . . . . . H .  1
   1 2 3 4 5 6 7 8
```

COMPUTADORA: 7
HUMANO: 8

	1	2	3	4	5	6	7	8	
8	.		.		.		.	K	8
7		.		.		.		.	7
6	.		.		.		.	C	6
5		.		.		.		.	5
4	.		.		.		.	H	4
3	$	.		.	H	.	H	.	3
2	.		.		.	$	.	H	2
1		.	$	.		.		.	1
	1	2	3	4	5	6	7	8	

COMPUTADORA: 8
HUMANO: 9

	1	2	3	4	5	6	7	8	
8	.		.	K	.		.		8
7		.		.	K	.		.	7
6	.		.	H	.	K	.		6
5		.		.		.		.	5
4	.		.		.	C	.		4
3		.	$	.	$	.		.	3
2	.		.		.		.		2
1		.		.		.		.	1
	1	2	3	4	5	6	7	8	

COMPUTADORA: 9
HUMANO: 10

	1	2	3	4	5	6	7	8	
8	.		.	K	.		.		8
7		.	H	.		.		.	7
6	.		.		.		.		6
5		.	$	.		.		.	5
4	.		.		.		.		4
3		.		.		.	K	.	3
2	.	$	.		.		.		2
1		.		.		.		.	1
	1	2	3	4	5	6	7	8	

COMPUTADORA: 11
HUMANO: 10

	1	2	3	4	5	6	7	8	
8	.		.		.		.		8
7		.		.		.		.	7
6	.	K	.	$	.		.		6
5		.		.		.		.	5
4	.		.		.		.		4
3		.		.		.		.	3
2	.	$	.		.		.		2
1		.		.		.		.	1
	1	2	3	4	5	6	7	8	

Y este es el listado de DAMAS:

```
   10 REM Damas
   20 GOSUB1010
   30 REM BORRA LA LINEA 50 PARA QUE
   40 REM MUEVA PRIMERO EL HUMANO
   50 GOTO90
   60 REM ****************************************
************************
   70 GOSUB600
   80 GOSUB760
   90 GOSUB600
  100 GOSUB130
  110 GOTO70
  120 REM ****************************************
************************
  130 FOR X =1 TO 10:S(X)=0:NEXTX
  140 SC=0:A=89
  150 A=A-1
  160 IFQ(A)<>C AND Q(A)<>CK THEN240
  170 B=0:IF A<29 THEN B=2
  180 B=B+1
  190 M=A+N(B)
  200 IFM>88 OR M<11 THEN240
  210 IF (Q(M)=H OR Q(M)=HK) AND Q(M+N(B))=E THEN
280
  220 IF Q(M)=E THEN IF (Q(M-11)<>H AND Q(M-11)<>H
K) THEN IF (Q(M-9)<>H AND Q(M-9)<>HK) AND Q(M+9)<>
HK THEN IF ((Q(M+22)<>HK OR Q(M+18)<>HK) AND (Q(M+
9)<>C OR Q(M+9)<>CK OR Q(M+11)=C OR Q(M+11)=CK)) A
ND Q(M+11)<>HK THEN GOSUB 400
  230 IF B<2 OR (Q(A)=CK AND B<4) THEN 180
  240 IFA>11 THEN 150
  250 FL=0:IFQ(22)=C OR Q(24)=C OR Q(26)=C OR Q(28
)=C THEN GOSUB 1270
  260 IF FL=1 THEN 570
  270 GOTO420
  280 Q(M+N(B))= Q(A): Q(M)=E:Q(A)=E
  290 CO=CO+1
  300 GOSUB600
  310 A=M +N(B)
```

```
320 B=0
330 B=B+1
340 IF (A+2*N(B)<11 OR A+2*N(B)>88) AND B<4 THEN
330
350 M=A+N(B)
360 IFQ(M)=C AND B>3 THEN RETURN
370 IF(Q(M)=H OR Q(M)=HK) AND Q(M+N(B))=E THEN 2
80
380 IF B<2 OR (Q(A)=CK AND B<4)THEN330
390 RETURN
400 IF SC <10 THEN SC =SC+1
410 S(SC)=100*A+B+20:RETURN
420 IF SC =0 THEN 470
430 XC=INT(RND(-1)*SC)+1
440 A=INT (S(XC)/100)
450 M=A+N(S(XC)- 100*A-20)
460 GOTO570
470 SC=SC+1:A=INT(RND(-1)*88)+1
480 IFQ(A)<>C AND Q(A)<> CK THEN 550
490 B=0
500 B=B+1
510 M=A+N(B)
520 IFM>88 OR M<11 THEN 540
530 IFQ(M)=E THEN 570
540 IF B<2 ORQ(A) =CK AND B<4 THEN 500
550 IFSC<300 THEN470
560 PRINT:PRINT" RECONOZCO QUE HAS GANADO":END
570 Q(M)=Q(A):Q(A)=E
580 RETURN
590 REM ****************************************
********************
600 CLS:PRINT:PRINT
610 PRINT "       COMPUTADORA:"CO
620 PRINT "             HUMANO:"HU:PRINT:PRINT
630 PRINT "     1 2 3 4 5 6 7 8"
640 FORF=8OTO10 STEP -10
650 PRINTF/10;" ";
660 FORG=1TO8:PRINTCHR$(Q(F+G));" ";:NEXTG
670 PRINTF/10:NEXTF
680 PRINT "     1 2 3 4 5 6 7 8"
690 IF CO=12 OR HU=12 THEN710
700 RETURN
710 IF HU =12 THEN PRINT "HAS VENCIDO"
```

```
  720 IF CO =12 THEN PRINT "HAS SIDO DERROTADO"
  730 PRINT " GRACIAS POR EL JUEGO":END
  740 REM ****************************************
***********************
  750 REM 99 PARA RENDIRSE
  760 PRINT:PRINT
  770 PRINT "CUAL ES TU MOVIMIENTO?"
  780 INPUT "DE ";A
  790 IFA=99 THEN GOTO730
  800 IFQ(A)<>H AND Q(A)<>HK THEN 780
  810 INPUT"A ";B
  820 IFQ(B)<>E THEN810
  830 Q(B)=Q(A):Q(A)=E
  840 REM ****************************************
**********************
  850 FORT=11 TO 17:IF Q(T) =C THEN Q(T)=CK
  860 NEXT T
  870 FOR T=82 TO 88:IFQ(T)=H THEN Q(T)= HK
  880 NEXT T
  890 REM ****************************************
***********************
  900 IF ABS(A-B)<12 THEN RETURN
  910 TY=RND(-1)
  920 IF TY<.3 THEN PRINT"BUEN MOVIMIENTO":GOSUB13
40
  930 IF TY>.7 THEN PRINT" ME HAS COGIDO!":GOSUB13
40
  940 HU=HU+1:Q((A+B)/2)=E:GOSUB600
  950 FOR T=82 TO 88:IFQ(T)=H THEN Q(T)=HK
  960 NEXTT
  970 PRINT:INPUT"PUEDES SALTAR DE NUEVO (S / N)";
A$
  980 IFA$<>"s" AND A$<>"S" THEN RETURN
  990. A=B:GOTO800
 1000 REM ****************************************
***********************
 1010 REM INICIALIZACION
 1020 CLS:PRINT "APRIETA CUALQUIER TECLA":N=1
 1030 N=N+1:IF INKEY$ ="" THEN 1030
 1040 CLS
 1050 PRINT "UN MOMENTO, POR FAVOR"
 1060 DIMQ(99),N(4),S(10)
 1070 H=ASC("H"):HK=ASC("K")
```

```
 1080 C=ASC("C"):CK=ASC("$")
 1090 E=32:OF=-99
 1100 FORM=1TO99:Q(M)=OF:NEXTM
 1110 FORM=1TO64
 1120 READD:READG
 1130 Q(D)=G:NEXTM
 1140 DATA 81,46,82,67,83,46,84,67,85,46,86,67,87,
46
 1150 DATA 88,67,71,67,72,46,73,67,74,46,75,67,76,
46
 1160 DATA 77,67,78,46,61,46,62,67,63,46,64,67
 1170 DATA 65,46,66,67,67,46,68,67,51,32,52,46
 1180 DATA 53,32,54,46,55,32,56,46,57,32,58,46
 1190 DATA 41,46,42,32,43,46,44,32,45,46,46,32
 1200 DATA 47,46,48,32,31,72,32,46,33,72,34,46,35,
72
 1210 DATA 36,46,37,72,38,46,21,46,22,72,23,46,24,
72
 1220 DATA 25,46,26,72,27,46,28,72,11,72,12,46,13,
72
 1230 DATA 14,46,15,72,16,46,17,72,18,46
 1240 FOR M=1TO 4:READ X:N(M)=X:NEXT M
 1250 DATA -11, -9, 11, 9
 1260 CO=0:HU=0:RETURN
 1270 IFQ(22)=C AND Q(11)=E THEN A=22:M=11:FL=1:RE
TURN
 1280  IFQ(22)=C AND Q(13)=E THEN A=22:M=13:FL=1:R
ETURN
 1290  IFQ(24)=C AND Q(13)=E THEN A=24:M=13:FL=1:R
ETURN
 1300  IFQ(24)=C AND Q(15)=E THEN A=24:M=15:FL=1:R
ETURN
 1310  IFQ(26)=C AND Q(15)=E THEN A=26:M=15:FL=1:R
ETURN
 1320  IFQ(26)=C AND Q(17)=E THEN A=26:M=17:FL=1:R
ETURN
 1330 RETURN
 1340 FORO=1 TO 1000:NEXTO :RETURN
```

SHOGUN

El SHOGUN está basado en un juego japonés llamado HASAMI SHOGI. En este juego, que se juega generalmente en una esquina de un tablero «GO», los jugadores pueden mover en cualquier dirección (hacia adelante, hacia atrás o hacia los lados) siempre que sea en línea recta. No se permite mover en diagonal. Asimismo, se puede saltar sobre las fichas (una de cada vez y siempre en línea recta) ya sea ésta de uno mismo o del contrario.

La ficha que se salta no se retira del tablero. El modo de comer las fichas del contrincante es *apresarlas* entre dos fichas propias. Es decir, el jugador humano ha de conseguir poner una ficha suya a cada lado de una ficha del ordenador (excepto en diagonal), en cuyo caso esta ficha se considera comida y se retira del tablero. Si una ficha tiene posibilidad de moverse entre dos fichas del contrincante, no puede ser comida por éste.

No existen jugadas dobles. Es decir, no se puede volver a mover después de comer. Originalmente el juego acababa cuando uno de los contrincantes había conseguido comer todas las fichas del otro, pero como esto supone un juego lento y largo y las partidas contra el ordenador podían eternizarse, decidí escribir un programa en el que el ganador fuera el que primero consiguiera comer seis fichas a su contrincante.

Aprenderá a jugar sencillamente viendo cómo se mueven las fichas. No se olvide de que se puede saltar sobre las fichas de uno mismo o sobre las del contrario. Es una buena táctica para conseguir poner las fichas en el centro del tablero casi al inicio del juego, lo que hará un juego más divertido.

Este es el listado del SHOGUN:

```
   10 REM Shogun
   20 REM ASEGURATE DE TRABAJAR CON MAYUSCULAS
   30 GOSUB 790
   40 GOSUB90
   50 GOSUB460
   60 GOSUB630
   70 GOSUB460
   80 GOTO40
   90 REM CAPTURA
  100 A=99
  110 IFA(A) <> C THEN 190
  120 IFA(A-10)=V THENIFA(A-9)=H THEN IFA(A-8)=C T
HENB=A-10:GOTO350
  130 IFA(A-10)=V THENIFA(A-11)=H THEN IFA(A-12)=C
 THENB=A-10:GOTO350
  140 IFA(A-10)=V THENIFA(A+11)=H THEN IFA(A+12)=C
 THENB=A-10:GOTO350
  150 B=1
  160 IFA+2*C(B)<11 OR A+2*C(B)>99 THEN GOTO180
  170 IF A(A+C(B))=V AND A(A+2*C(B))=H AND A(A+3*C
(B)) = C THEN A(A+2*C(B))=V:  CS=CS+1:GOTO340
  180 IF B<4 THEN B=B+1:GOTO160
  190 IF A>11 THEN A=A-1:GOTO110
  200 REM NO HAY CAPTURA
  210 CUENTA=0
  220 CUENTA=CUENTA+1
  230 A=RND(-1)*89+11
  240 IFA(A)=C THEN 270
  250 IF CUENTA <200 THEN 220
  260 PRINT "Maestro del Shogun":PRINT "Yo te doy
la victoria!":END
  270 B=1
  280 IFA+2*C(B)<11 THEN300
  290 IF(A(A+C(B))=C OR A(A+C(B)) =H) AND A(A+2*C(
B)) = V THEN B=A + 2*C(B):GOTO350
  300 IF A(A+C(B))=V THEN330
  310 IFB<4 THENB=B+1:GOTO280
  320 GOTO250
  330 REM MUEVE LA COMPUTADORA
  340 B=A+C(B)
```

```
350 B1=B-10*(INT(B/10))
360 A(B)=C:A(A)=V
370 IFB1>7 THEN 390
380 IF A(B+1) =H AND A(B+2)=C THENA(B+1)=V:CS=CS
+1
390 IFB1<3 THEN410
400 IF A(B-1) =H AND A(B-2)=C THENA(B-1)=V:CS=CS
+1
410 IF A>89 THEN 430
420 IF A(B+10) =H AND A(B+20)=C THENA(B+10)=V:CS
=CS+1
430 IF A<29 THEN RETURN
440 IF A(B-10) =H AND A(B-20)=C THENA(B-10)=V:CS
=CS+1
450 RETURN
460 REM IMPRESION DEL TABLERO
470 LOCATE0,0:REM PONGA HOME AQUI SI SU COMPUTAD
ORA TIENE ESTE COMANDO
480 PRINT TAB(15);"1 2 3 4 5 6 7 8 9"
490 FORM=90 TO10 STEP-10
500 PRINT TAB(13) CHR$(M/10+64);" ";
510 FOR N=1 TO 9
520 PRINT CHR$(A(M+N));" ";
530 NEXT
540 PRINT CHR$(M/10+64)
550 NEXT
560 PRINT TAB(15);"1 2 3 4 5 6 7 8 9"
570 PRINT:PRINT ,"COMPUTADORA:"CS
580 PRINT ,"HUMANO:"HS
590 IF CS>6 OR HS>6 THEN610
600 RETURN
610 IF CS > HS THEN PRINT:PRINT "YO GANO!":END
620 PRINT:PRINT "TU GANAS!":END
630 REM MOVIMIENTO DEL JUGADOR
640 INPUT "Desde (letra,No)";A$
650 IFA$="S" THEN END
660 IF LEN (A$)<>2 THEN 640
670 PRINT "Desde "A$" a ";:INPUT B$
680 IF LEN(B$)<>2 THEN 670
690 A=10*(ASC(A$)-64)+VAL(RIGHT$(A$,1))
700 B=10*(ASC(B$)-64)+VAL(RIGHT$(B$,1))
710 Y=VAL(RIGHT$(B$,1))
720 A(B)=H:A(A)=V
```

```
  730 IFA(B+1)=C AND A(B+2)=H AND Y<=7 THEN A(B+1)
=V:HS=HS+1
  740 IFA(B-1)=C AND A(B-2)=H AND Y>=3 THEN A(B-1)
=V:HS=HS+1
  750 IFB>79 THEN770
  760 IFA(B+10)=C AND A(B+20)=H THENA(B+10)=V:HS=H
S+1
  770 IFB>=31 THEN IF A(B-10)=C AND A(B-20)=H THEN
 A(B-10)=V:HS=HS+1
  780 RETURN
  790 REM INICIALIZACION
  800 REM PREPARA EL GENERADOR DE NUMEROS ALEATORI
OS
  810 PRINT "APRIETA CUALQUIER TECLA"
  820 N=1
  830 N=N+1
  840 IF INKEY$="" THEN830
  850 RANDOMIZE
  870 CLS
  880 DIM A(129), C(4)
  890 H=72:C=67:V=42
  900 FORZ=11TO29
  910 IFZ=20 THENZ=21
  920 A(Z)=H
  930 NEXT
  940 FORZ=31 TO 79
  950 IF 10*INT(Z/10)=Z THEN Z=Z+1
  960 A(Z)=V
  970 NEXT
  980 FORZ=81 TO 99
  990 IFZ=90 THEN Z=91
 1000 A(Z)=C
 1010 NEXT
 1020 HS=0
 1030 CS=0
 1040 FORZ=1TO4
 1050 READC(Z)
 1060 NEXT
 1070 DATA -10, -1, 1, 10
 1080 GOSUB460
 1090 RETURN
```

AWARI

El AWARI es uno más de esos muchos juegos que se juegan con guijarros en el suelo a lo largo y ancho del mundo, de Africa a Filipinas, y que ahora entrará en su hogar traído por un inteligente contrincante: su ordenador.

Como puede ver en la muestra que le ofrecemos a continuación, al inicio del juego cada jugador tiene frente a sí seis hoyos (las letras de A a F y de L a G). Cada hoyo contiene tres guijarros. Escogiendo cualquiera de los hoyos de su lado, el jugador recoge los tres guijarros de éste y posteriormente procede a lanzarlos en el sentido de las agujas del reloj sembrando con uno de ellos cada hoyo que encuentre a su paso. No se siembran los hoyos que están en los extremos del tablero, los que parten de cero.

Si el último de los tres guijarros cae enfrente de un hoyo vacío, en ese caso todos los guijarros del hoyo en que ha caído pasan a este jugador, quien se los lleva a su «casa». La casa del jugador humano es el hoyo que parte de cero en el lado izquierdo del tablero, la del ordenador es el de la derecha.

El juego continúa hasta que no quedan guijarros en ninguno de los hoyos a ambos lados, de modo que los jugadores ya no pueden mover. Gana aquel que consiga tener más guijarros en «casa». El ordenador juega bastante bien, pero con algo de práctica puede llegar a vencerle. No espere ganarle al principio. Si no quiere ver la jugada del ordenador antes de que la haga, suprima el INPUT Z en la línea 270. Asimismo, puede borrar el bucle de la línea 40 con lo que conseguirá el máximo de rapidez (aunque, de hecho, el ordenador juega muy deprisa aun en su forma presente).

Este es un ejemplo de juego contra el ordenador:

```
   A  B  C  D  E  F
   3  3  3  3  3  3
0                    0
   3  3  3  3  3  3
   L  K  J  I  H  G

Me movere desde C
?
```

```
   A  B  C  D  E  F
   3  3  0  4  4  4
0                    0
   3  3  3  3  3  3
   L  K  J  I  H  G

Con que agujero quieres empezar?
```

```
   A  B  C  D  E  F
   3  3  0  4  4  4
0                    0
   3  4  4  4  0  3
   L  K  J  I  H  G
```

```
   A  B  C  D  E  F
   3  3  0  0  0  5
0                    5
   3  4  4  4  1  4
   L  K  J  I  H  G
```

```
   A  B  C  D  E  F
   0  1  1  2  1  5
8                    7
   4  1  1  5  0  0
   L  K  J  I  H  G

Con que agujero quieres empezar?
```

```
   A  B  C  D  E  F
   1  2  1  2  1  5
8                    7
   5  2  2  0  0  0
   L  K  J  I  H  G

Me movere desde A
?
```

```
   A  B  C  D  E  F
   0  3  1  2  1  5
8                    7
   5  2  2  0  0  0
   L  K  J  I  H  G
```

```
   A  B  C  D  E  F
   1  4  2  3  2  5
8                    7
   0  2  2  0  0  0
   L  K  J  I  H  G
```

Con que agujero quieres empezar?L

```
   A  B  C  D  E  F
   3  5  1  0  0  0
9                   15
   0  0  0  1  2  0
   L  K  J  I  H  G
```

```
   A  B  C  D  E  F
   3  0  2  1  1  0
9                   16
   0  0  0  1  2  1
   L  K  J  I  H  G
```

Me movere desde B
?

Con que agujero quieres empezar?I

```
   A  B  C  D  E  F
   3  0  0  1  1  0
11                  16
   0  0  1  0  2  1
   L  K  J  I  H  G
```

```
   A  B  C  D  E  F
   3  0  0  1  0  0
11                  16
   0  0  1  0  2  1
   L  K  J  I  H  G
```

Me movere desde E
?

```
   A  B  C  D  E  F          A  B  C  D  E  F
   0  1  0  0  0  0          0  0  1  0  0  0
14                   17  14                   20
   0  0  3  1  0  0          0  0  0  1  0  0
   L  K  J  I  H  G          L  K  J  I  H  G


Me movere desde B        Con que agujero quieres empezar?I
?


                   A  B  C  D  E  F
                   0  0  0  0  0  0
                15                   20
                   0  0  1  0  0  0
                   L  K  J  I  H  G


                ESTE HA SIDO EL FINAL DEL JUEGO

                Y YO SOY  EL VENCEDOR!

                MI PUNTUACION FUE  20
                     Y LA TUYA FUE 15
```

Y este es el listado del AWARI:

```
10 REM Awari
20 GOSUB 930:REMInicializacion
30 GOSUB770:REM Imprime el tablero
40 FORP=1TO500:NEXTP
```

```
 50 GOSUB160:REM Movimiento de la maquina
 60 GOSUB 770:REM Imprime el tablero
 70 GOSUB 620:REM Movimiento del humano
 80 CW=0:HW=0
 90 FORC=1 TO 12
100 IFC<7 THEN CW=CW+A(C)
110 IFC>6 THEN HW=HW+A(C)
120 NEXTC
130 IFCW=0 OR HW=0 THEN510
140 GOTO30
150 REM ****************************************
************************
160 REM Movimiento de la maquina
170 GM=0:C=0
180 C=C+1
190 IFA(C)=0 THEN 180
200 Z=C+A(C)
210 IFZ>12 THENZ=Z-12
220 IFZ>6 THEN IF A(Z-6)<>0 AND A(Z)=0 AND A(Z-6
)> GM THEN GM=C
230 IFZ<7 THEN IF A(Z+6)<>0 AND A(Z)=0 AND A(Z+6
)> GM THEN GM=C
240 IFC<6 THEN 180
250 IF GM=0 THEN370
260 C=GM
270 PRINT"Yo movere desde "CHR$(64+C):INPUT Z$
280 FORZ=C TO C+A(C)
290 IFZ>12 THEN A(Z-12)=A(Z-12)+1
300 IFZ<13 THENA(Z)=A(Z)+1
310 NEXTZ
320 Z=C+A(C)-1:IFZ>12 THEN Z=Z-12
330 A(C)=0
340 B(2)=B(2)+A(13-Z):A(13-Z)=0
350 RETURN
360 REM ****************************************
********************
370 REM Movimiento sin puntuacion
380 W=0
390 W=W+1
400 C=INT(RND(-1)*6)+1
410 IF A(C)<>0 THEN 440
420 IF W<20 THEN 390
430 GOTO510
```

```
440 PRINT"Me movere desde "CHR$(64+C):INPUT Z$
450 FORZ=C TO C+A(C)
460 IF Z<13 THEN A(Z) = A(Z)+1
470 IF Z>12 THEN A(Z-6)=A(Z-6)+1
480 NEXT Z
490 A(C)=0 :GOTO350
500 REM ****************************************
*******************
510 REM Fin del juego
520 GOSUB770
530 PRINT:PRINT"ESTE HA SIDO EL FINAL DEL JUEGO"
540 PRINT
550 IF B(1)>B(2) THEN PRINT"ERES EL VENCEDOR!!"
560 IFB(1)<B(2) THEN PRINT"Y YO SOY  EL VENCEDOR
!"
570 IFB(1)=B(2)THEN PRINT"CREO QUE ES UN AHOGADO
!"
580 PRINT:PRINT"MI PUNTUACION FUE "B(2)
590 PRINT"     Y LA TUYA FUE"B(1)
600 END
610 REM ****************************************
*********************
620 REM Movimiento del humano
630 INPUT" Con que agujero quieres empezar ";A$
640 B=ASC(A$)-64
650 IFB<7 OR B>12 THEN630
660 CO=B:Z=B+A(B):IF Z>12 THENZ=Z-12
670 M=A(Z)
680 FORZ=B TO B+A(B)
690 IF Z>12 THEN A(Z-12)=A(Z-12)+1
700 IFZ<13 THENA(Z)=A(Z)+1
710 NEXT Z
720 Z=B+A(B)-1:IFZ>12 THEN Z=Z-12
730 IFM=0 THENB(1)=B(1)+A(13-Z):A(13-Z)=0
740 A(CO)=0
750 RETURN
760 REM ****************************************
**********************
770 REM Imprime el tablero
780 CLS:PRINT:PRINT:PRINT
790 PRINT"    A  B  C  D  E  F":PRINT "   ";
800 FORC=1TO6
810 PRINTA(C);
```

```
 820 NEXTC
 830 PRINT:PRINTB(1)"                    "B(2):PRINT
"    ";
 840 FORC=12 TO 7 STEP-1
 850 PRINT A(C);
 860 NEXTC
 870 PRINT:PRINT "    L  K  J  I  H  G"
 880 PRINT:PRINT
 890 RETURN
 900 PRINT "ME MOVERE DESDE "CHR$(64+GM)
 910 C=GM
 920 REM ****************************************
**********************
 930 REM INICIALIZACION
 940 CLS
 950 RANDOMIZE
 960 DIM A(12), B(2)
 970 FORC=1 TO 12
 980 A(C)=3
 990 NEXTC
1000 RETURN
```

KNIGHTSBRIDGE

Es éste uno de los pocos juegos que puedo decir que he inventado en su totalidad (aunque, por supuesto, la idea de ponerle el nombre de un barrio de Londres la tomé de los dos inventores del Kensington). Tengo la costumbre de leer libros de juegos para sacar ideas y muchas veces cuando me voy a pasear por los parques londinenses llevo un libro conmigo para leer cuando me canso de andar y me apetece descansar un rato. Este juego se me ocurrió una tarde de domingo mientras paseaba por «Ruislip Lido» pensando en algo que acababa de leer en un libro. Se decía en éste que el ajedrez, tal como se conoce ahora, es la evolución de un juego indio en el que el lanzamiento de dados dictaba la ficha que había que mover.

Partiendo de este punto, pensé en un juego en el que todas las fichas fueran caballos de ajedrez y en el que se tiraran los dados de cada vez para saber qué ficha iba a mover. El Knightsbridge se desarrolló a partir de esta idea.

Se juega en un tablero de siete por siete. El ordenador le indica qué ficha tiene que mover (cada ficha se indica introduciendo un número de dos dígitos que es el número de la casilla resultante, en la que se encuentran uno de los números situados en un lateral del tablero y otro de los situados en la parte superior de éste), y le deja decidir a dónde va a moverla. Todas las fichas se mueven como los caballos de ajedrez y antes de permitirle hacer la jugada, el ordenador la comprueba para ver si es correcta.

El modo de comer es caer en una casilla en la que haya una ficha del enemigo. El primer jugador que logre comer cinco de las siete fichas del contrario, es el ganador. No se dan saltos múltiples.

Encontrará que el ordenador juega rápido y bien.

Y aquí tenemos al inventor del juego siendo derrotado por su propia invención:

```
Mi puntuacion es  0
Y la tuya es  0

 1 2 3 4 5 6 7
 -------------
7 C . C C C C C 7
6 . . . C . . . 6
5 . . . . . . . 5
4 . . . . . . . 4
3 . . . . . . . 3
2 . . . . . . . 2
1 H H H H H H H 1
 -------------
 1 2 3 4 5 6 7

Debes mover la ficha sobre 17
?36
```

```
Mi puntuacion es  0
Y la tuya es  0

 1 2 3 4 5 6 7
 -------------
7 C . C . C C C 7
6 . . . . . C . 6
5 . . . . . C . 5
4 . . . . . . . 4
3 . . . H . H . 3
2 . . . . . . . 2
1 H H H H . H . 1
 -------------
 1 2 3 4 5 6 7

Debes mover la ficha sobre 34
?53
```

```
Mi puntuacion es  0
Y la tuya es  0

 1 2 3 4 5 6 7
 -------------
7 C . C . C C C 7
6 . . . C . C . 6
5 . . . . . . . 5
4 . . . . . . . 4
3 . . . . . H . 3
2 . . . . . . . 2
1 H H H H H H . 1
 -------------
 1 2 3 4 5 6 7

Debes mover la ficha sobre 15
?34
```

```
Mi puntuacion es  0
Y la tuya es  0

 1 2 3 4 5 6 7
 -------------
7 C . . . C C C 7
6 . . . . C C . 6
5 . . H . . C . 5
4 . . . . . . . 4
3 . . . . . H . 3
2 . . . . . . . 2
1 H H H H . H . 1
 -------------
 1 2 3 4 5 6 7

Debes mover la ficha sobre 12
?33
```

```
Mi puntuacion es  3
Y la tuya es  2

  1 2 3 4 5 6 7
  -------------
7 . . . . C C C 7
6 . . C . . . . 6
5 . . . H . . . 5
4 . . . . . . . 4
3 . . . . . H . 3
2 . . H . . C . 2
1 . . . H . . . 1
  -------------
  1 2 3 4 5 6 7

Debes mover la ficha sobre 23
?44
```

```
Mi puntuacion es  4
Y la tuya es  2

  1 2 3 4 5 6 7
  -------------
7 . . . . C C C 7
6 . . . . . . . 6
5 . . . H . . . 5
4 . . . C . . . 4
3 . . . . . H . 3
2 . . . . . C . 2
1 . . . H . . . 1
  -------------
  1 2 3 4 5 6 7

Debes mover la ficha sobre 14
?26
```

```
Mi puntuacion es  4
Y la tuya es  3

  1 2 3 4 5 6 7
  -------------
7 . . . . C . C 7
6 . . . . . . . 6
5 . . . H C . . 5
4 . . . C . . . 4
3 . . . . . H . 3
2 . . . . . H . 2
1 . . . . . . . 1
  -------------
  1 2 3 4 5 6 7

Debes mover la ficha sobre 36
?55
```

```
Mi puntuacion es  5
Y la tuya es  4

1 2 3 4 5 6 7
-------------
7 . . . . . . 7
6 . . . . . . 6
5 . . C H . . 5
4 . . C . . . 4
3 . . . . . . 3
2 . . . . H . 2
1 . . . . . . 1
-------------
1 2 3 4 5 6 7

ESTA VICTORIA ES LA PRIMERA DENTRO DE
 UN PLAN GLOBAL PARA CONQUISTAR LA TIERRA!
ACEPTO TU DESEO DE RENDIRTE
```

Aquí está el listado de KNIGHTSBRIDGE:

```
  10 REM Knightsbridge
  20 GOSUB 760:REM inicializacion
  30 GOSUB 490:REM Imprime tablero
  40 IF HU =5 OR CO=5 THEN 680:REM Fin del juego
  50 GOSUB 300:REM Mueve la computadora
  60 GOSUB 490:REM Imprime el tablero
  70 IF HU=5 OR CO=5 THEN 680:REM Fin del juego
  80 GOSUB 110:REM Mueve el jugador
  90 GOTO 30
 100 REM ****************************************
*************
```

```
110 REM Mueve el jugador
120 Q=0
130 M = INT(RND(-2)*66) + 11
140 Q = Q + 1
150 IF Q = 500 THEN 680
160 IF H(M) <> 72 THEN 130
170 PRINT "Debes mover la ficha sobre "M
180 INPUT N
190 IF N = 99 THEN Q = 500 : GOTO 680
200 REM *****Comprueba si el movimiento es legal
*****
210 P = 0
220 CT = 1
230 IF M + Z(CT) = N THEN P = 1
240 IF CT < 8 THEN CT = CT + 1 :GOTO 230
250 IF P = 0 THEN PRINT "Movimiento ilegal":GOTO
 180
260 IF H(N) = 67 THEN HU = HU + 1: PRINT"Bien ju
gado!":FOR R = 1 TO 500:NEXT R
270 H(M) = 46: H(N) = 72
280 RETURN
290 REM ****************************************
*************
300 REM Mueve la computadora
310 Q1 = 0
320 Q1 = Q1 + 1
330 K = INT(RND(-2)*66) + 11
340 IF Q1 = 500 THEN 680
350 IF H(K) <> 67 THEN 320
360 PRINT "Tengo que mover la ficha sobre "K
365 FOR Z=1 TO 1000:NEXTZ
370 W = 1
380 IF K + Z(W) < 11 OR K + Z(W) > 77 THEN 400
390 IF H(K + Z(W)) = 72 THEN PRINT "MUY BUENO...
":CO = CO + 1:FOR P=1 TO 200:NEXT P:GOTO 450
400 IF W < 8 THEN W = W + 1:GOTO 380
410 W = 1
420 IF (K + Z(W) < 11 OR K + Z(W) > 77) AND W <
8 THEN W = W + 1:GOTO 420
430 IF H(K + Z(W)) <> 46 AND W < 8 THEN W = W +
1:GOTO 430
440 IF W = 8 AND H(K + Z(W)) <> 46 THEN Q1 = 500
:GOTO 680
```

```
450 X = K:Y = K + Z(W)
460 H(X) = 46:H(Y) = 67
470 RETURN
480 REM ****************************************
******************
490 REM Imprime el tablero
500 CLS:PRINT:PRINT:PRINT
510 PRINT TAB(8);"Mi puntuacion es "CO
520 PRINT TAB(8);"Y la tuya es "HU
530 PRINT
540 PRINT TAB(8);"1 2 3 4 5 6 7"
550 PRINT TAB(8);"-------------"
560 FOR M = 70 TO 10 STEP -10
570 PRINT TAB(5);M / 10;
580 FOR N = 1 TO 7
590 PRINT CHR$(H(M + N));" ";
600 NEXT N
610 PRINT M / 10
620 NEXT M
630 PRINT TAB(8);"-------------"
640 PRINT TAB(8);"1 2 3 4 5 6 7"
650 PRINT
660 RETURN
670 REM ****************************************
*******************
680 REM Final del juego
690 GOSUB 490
700 IF HU = 5 THEN PRINT "ME HAS VENCIDO, HUMANO
"
710 IF CO = 5 THEN PRINT" ESTA VICTORIA ES LA PR
IMERA DENTRO DE":PRINT  "  UN PLAN GLOBAL PARA CON
QUISTAR LA TIERRA!"
720 IF Q = 500 THEN PRINT " ACEPTO TU DESEO DE R
ENDIRTE"
730 IF Q1 = 500 THEN PRINT " ME RINDO ANTE UN MA
ESTRO"
740 END
750 REM ****************************************
*******************
760 REM Inicializacion
770 CLS:PRINT "UN MOMENTO, POR FAVOR... HUMANO"
780 DIM H(99), Z(8)
790 X = 0:Q1 = 0:Q = 0
```

```
 800 RANDOMIZE
 810 HU = 0:CO = 0:REM Puntuaciones
 820 FOR A = 1 TO 99
 830 IF A > 77 OR A = 70 OR A = 60 OR A = 68 OR A
= 69 OR A = 50 OR A = 59 OR A = 40 OR A = 48 OR A
= 49 THEN 880
 840 IF A = 30 OR A = 38 OR A = 39 OR A = 20 OR A
= 28 OR A = 29 OR A = 18 OR A = 19 OR A < 11 THEN
880
 850 H(A) = 46
 860 IF A > 70 AND A < 78 THEN H(A) = 67
 870 IF A > 10 AND A < 18 THEN H(A) = 72
 880 NEXT A
 890 FOR A=1 TO 8:READ Z(A):NEXT A
 900 DATA -8, -21, -12, -19, 19, 12, 21, 8
 910 RETURN
```

REVERSI/OTHELLO

El último programa de este capítulo es el REVERSI, que a menudo también se llama OTHELLO. Inventado a finales del siglo pasado, se juega en un tablero normal de ocho por ocho. Cuando se juega en un tablero real, se utilizan fichas que son de un color diferente por cada lado. El juego comienza con cuatro fichas situadas en las casillas centrales.

Desde este momento, el jugador humano mueve situando una de sus fichas al lado de una ficha o fichas del ordenador y procurando que haya otra ficha suya al otro lado de aquella o aquellas. Cuando sucede esto, la ficha o fichas del ordenador se convierten en fichas del jugador humano.

Así es como funciona. Suponga que se diera una línea en la que las fichas estuvieran colocadas del modo siguiente:

OXXXX

... y que usted decidiera poner su ficha (la O) al final de la línea; de este modo:

OXXXXO

las fichas del ordenador se convertirían en fichas suyas, de modo que la composición de la línea después de la jugada sería:

OOOOOO.

El juego continúa hasta completar todas las casillas del tablero o hasta que ninguno de los dos jugadores pueda mover. Como se puede ver, la suerte va pasando a toda velocidad de un jugador a otro, ya que las líneas que se van creando (como las diagonales) pueden modificarse con una sola jugada.

Si en un momento determinado el jugador humano no puede mover, se lo dice al ordenador introduciendo un cero.

El Reversi es un juego muy divertido. En un artículo publicado en la revista *Creative Computing* (junio, 1981, pág. 188), David Levy era de la opinión de que es «uno de los mejores juegos que se hayan inventado nunca, simplemente porque teniendo unas reglas tan sencillas que un minuto es suficiente para aprenderlas, es, sin embargo, un juego que requiere con frecuencia años de aprendizaje antes de que uno llegue a dominarlo». Posteriormente observa que es un juego complejo, más complejo, por ejemplo, que las damas, aunque menos que el ajedrez.

Aunque el juego moderno se llama OTHELLO * y fue «inventado» en Tokio en 1971, por Goro Hasegwana, la única diferencia entre éste y el antiguo Reversi radica en la colocación de partida de las cuatro primeras fichas que ha de ser como se indica en el ejemplo gráfico que sigue a esta descripción.

Pero si volvemos a los años ochenta del siglo pasado, cuando se inventó el juego en Londres, nos encontramos con que son dos los caballeros —Lewis Waterman y John W. Mollete— los que afirman ser inventores del juego. Stephen Kimmel, en un artículo publicado en la revista *Creative Computing* (julio, 1981, pág. 94) dice que para él es Waterman el que cuenta con más pruebas y argumentos a su favor ya que fue a él a quien primero le publicaran los pormenores de éste en una serie de artículos aparecidos en la revista *The Queen* (que estaba especializada en «asuntos de interés para las damas»).

Independientemente de quién haya sido su creador, es un gran juego y este programa (basado en uno escrito por Graham Charlton) ofrece una fogosa defensa. Esta muestra es un ejemplo de las primeras jugadas de una partida cualquiera.

---

* Othello es marca registrada de Gabriel Industries, Inc., USA, y Mine Information, UK.

```
LA COMPUTADORA ES X
     EL HUMANO ES O

   1 2 3 4 5 6 7 8
1  . . . . . . . .  1
2  . . . . . . . .  2
3  . . . . . . . .  3
4  . . . X O . . .  4
5  . . . O X . . .  5
6  . . . . . . . .  6
7  . . . . . . . .  7
8  . . . . . . . .  8
   1 2 3 4 5 6 7 8

COMPUTADORA: 2

HUMANO: 2
```

```
LA COMPUTADORA ES X
     EL HUMANO ES O

   1 2 3 4 5 6 7 8
1  . . . . . . . .  1
2  . . . . . . . .  2
3  . . . . . . . .  3
4  . . . X O . . .  4
5  . . X X X . . .  5
6  . . . . . . . .  6
7  . . . . . . . .  7
8  . . . . . . . .  8
   1 2 3 4 5 6 7 8

COMPUTADORA: 4

HUMANO: 1
```

```
LA COMPUTADORA ES X
     EL HUMANO ES O

   1 2 3 4 5 6 7 8
1  . . . . . . . .  1
2  . . . . . . . .  2
3  . . . . . . . .  3
4  . . . X O . . .  4
5  . . X O X . . .  5
6  . . O . . . . .  6
7  . . . . . . . .  7
8  . . . . . . . .  8
   1 2 3 4 5 6 7 8

COMPUTADORA: 3

HUMANO: 3
```

```
LA COMPUTADORA ES X
     EL HUMANO ES O

   1 2 3 4 5 6 7 8
1  . . . . . . . .  1
2  . . . . . . . .  2
3  . . . . . . . .  3
4  . . . X O . . .  4
5  . . X O X . . .  5
6  . . X . . . . .  6
7  . . X . . . . .  7
8  . . . . . . . .  8
   1 2 3 4 5 6 7 8

COMPUTADORA: 5

HUMANO: 2
```

```
LA COMPUTADORA ES X
     EL HUMANO ES O

   1 2 3 4 5 6 7 8
1  . . . . . . . .  1
2  . . . . . . . .  2
3  . . . . . . . .  3
4  . . . X O . . .  4
5  . . X X X . . .  5
6  . . O X . . . .  6
7  . O X . . . . .  7
8  . . . . . . . .  8
   1 2 3 4 5 6 7 8

COMPUTADORA: 6

HUMANO: 3
```

```
LA COMPUTADORA ES X
     EL HUMANO ES O

   1 2 3 4 5 6 7 8
1  . . . . . . . .  1
2  . . . . . . . .  2
3  . . . . . . . .  3
4  . . O O O . . .  4
5  . . O X X . . .  5
6  . . O X . . . .  6
7  . O X . . . . .  7
8  . . . . . . . .  8
   1 2 3 4 5 6 7 8

COMPUTADORA: 4

HUMANO: 6
```

```
LA COMPUTADORA ES X
     EL HUMANO ES O

   1 2 3 4 5 6 7 8
1  . . . . . . . .  1
2  . . . . . . . .  2
3  . . X . . . . .  3
4  . . X X O . . .  4
5  . . X X X . . .  5
6  . . X X . . . .  6
7  . O X . . . . .  7
8  . . . . . . . .  8
   1 2 3 4 5 6 7 8

COMPUTADORA: 9

HUMANO: 2
```

```
LA COMPUTADORA ES X
     EL HUMANO ES O

   1 2 3 4 5 6 7 8
1  . . . . . . . .  1
2  . . . . . . . .  2
3  . . X . . . . .  3
4  . O O O O . . .  4
5  . . X X X . . .  5
6  . . X X . . . .  6
7  . O X . . . . .  7
8  . . . . . . . .  8
   1 2 3 4 5 6 7 8

COMPUTADORA: 7

HUMANO: 5
```

```
LA COMPUTADORA ES X
     EL HUMANO ES O

   1 2 3 4 5 6 7 8
1  . . . . . . . .  1
2  . . . . . . . .  2
3  . . X . . . . .  3
4  . X O O O . . .  4
5  X . O X X . . .  5
6  . . O O . . . .  6
7  . X O . O . . .  7
8  X . O . . . . .  8
   1 2 3 4 5 6 7 8

COMPUTADORA: 7

HUMANO: 9
```

```
LA COMPUTADORA ES X
     EL HUMANO ES O

   1 2 3 4 5 6 7 8
1  . . . . . . . .  1
2  . . . . . . . .  2
3  . . X . . . . .  3
4  . X O O O . . .  4
5  X . X X X . . .  5
6  . . O X . . . .  6
7  . X O . X . . .  7
8  X . O . . X . .  8
   1 2 3 4 5 6 7 8

COMPUTADORA: 11

HUMANO: 6
```

```
LA COMPUTADORA ES X
     EL HUMANO ES O

   1 2 3 4 5 6 7 8
1  . . . . . . . .  1
2  . . . . . . . .  2
3  . . X . . . . .  3
4  . X O O O . . .  4
5  X . X O O . . .  5
6  . . O O O . . .  6
7  . X O . X . . .  7
8  X . O . . X . .  8
   1 2 3 4 5 6 7 8

COMPUTADORA: 8

HUMANO: 10
```

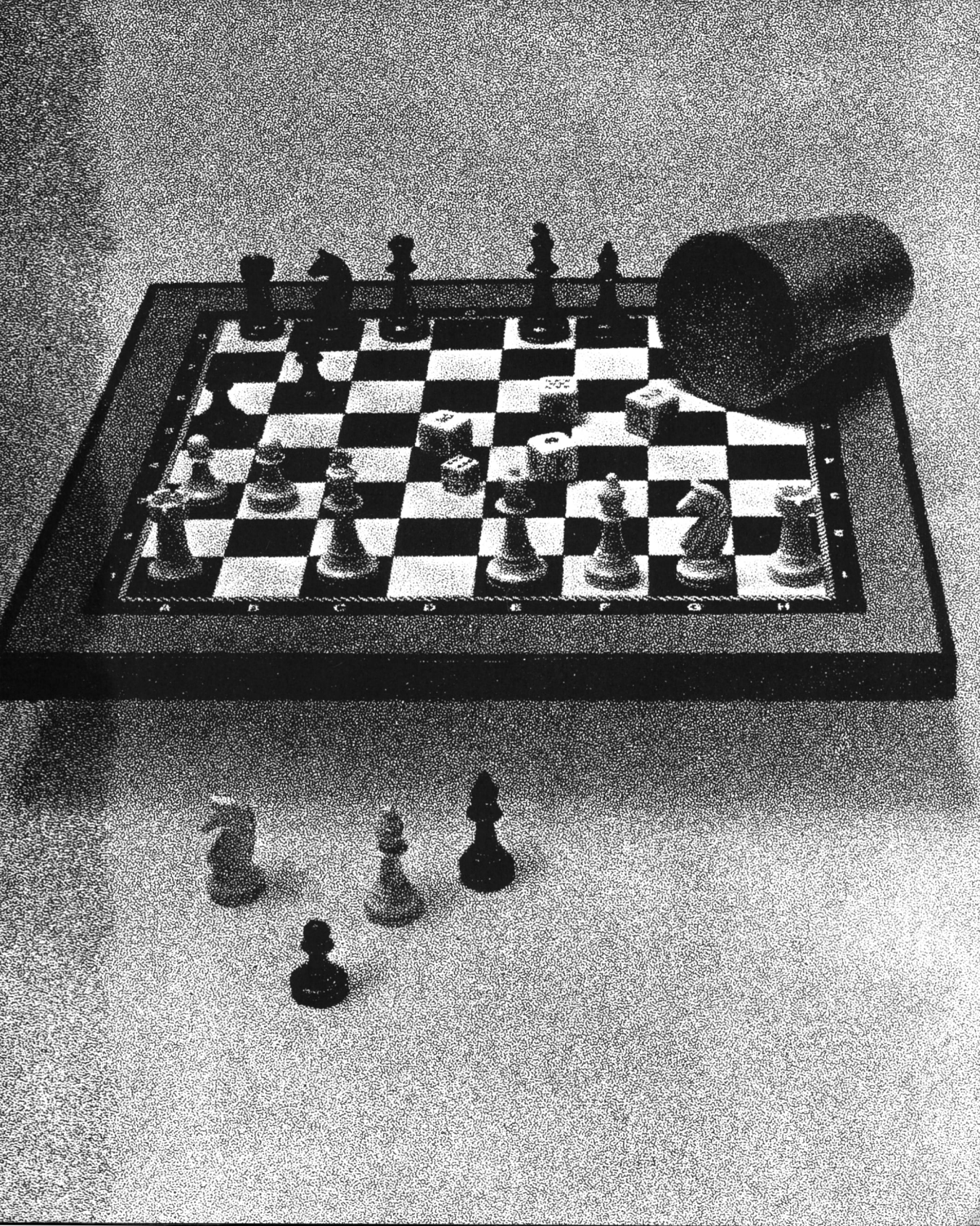
```
LA COMPUTADORA ES X
     EL HUMANO ES O

   1 2 3 4 5 6 7 8
1  . . . . . . . .  1
2  . . . . . . . .  2
3  . . X . X . . .  3
4  . X O X X . . .  4
5  X . X O X . . .  5
6  . . O O X . . .  6
7  . X O . X . . .  7
8  X . O . . X . .  8
   1 2 3 4 5 6 7 8

COMPUTADORA: 13

HUMANO: 6
```

Y aquí tenemos el listado de OTHELLO:

```
 10 REM Reversi/Otelo
 20 GOTO740
 30 PRINT "MI MOVIMIENTO..."
 40 S=O:T=X:H=O
 50 FOR A=2 TO 9:FORB=2 TO 9
 60 IF A(A,B)<>46 THEN 210
 70 Q=O
 80 FORC=-1 TO1:FORD=-1 TO1
 90 K=O:F=A:G=B
100 IFA(F+C,G+D)<>S THEN 130
110 K=K+1:F=F+C:G=G+D
120 GOTO100
130 IFA(F+C,G+D)<>T THEN 150
140 Q=Q+K
150 NEXTD:NEXTC
160 IFA=2 OR A=9OR B=2 OR B=9THEN Q=Q*2
170 IFA=3 OR A=8 OR B=3OR B=8 THEN Q=Q/2
180 IF (A=2 OR A=9)AND (B=3 OR B=8)OR (A=3 OR A=
8)AND(B=2 OR B=9) THEN Q=Q/2
190 IF Q<H OR (RND(-1)<.3 AND Q=H) THEN 210
200 H=Q:M=A:N=B
210 NEXTB:NEXTA
220 IFH=O AND R=O THEN 690
230 IF H=O THEN 250
240 GOSUB 580
250  GOSUB370
260 INPUT "CUAL ES TU MOVIMIENTO ";R
270 REM SELECCIONA O PARA PASAR
280 S=X:T=O:REM LETRA O
290 IF R=O THEN 350
300 IF R<11 OR R>88 THEN 260
310 R=R+11
320 M=INT(R/10)
330 N=R-10*M
340 GOSUB580
350 GOSUB370
360 GOTO 30
370 REM Imprime el tablero
380 C=O:H=O
```

```
390 CLS:REM OR 'HOME'
400 PRINT
410 PRINT "      LA COMPUTADORA ES X"
420 PRINT "            EL HUMANO ES O"
430 PRINT
440 PRINT TAB(5);"1 2 3 4 5 6 7 8"
450 FORB=2 TO 9:PRINT B-1;"  ";
460 FOR D=2 TO9
470 PRINT CHR$(A(B,D));" ";
480 IFA(B,D)=X THEN C=C+1
490 IFA(B,D)=O THENH=H+1
500 NEXTD
510 PRINTB-1
520 NEXTB
530 PRINT TAB(5);"1 2 3 4 5 6 7 8"
540 PRINT:PRINT
550 PRINT"COMPUTADORA:"C
560 PRINT:PRINT"HUMANO:"H
570 RETURN
580 FORC=-1TO1:FORD=-1TO1
590 F=M:G=N
600 IF A(F+C,G+D)<> S THEN630
610 F=F+C:G=G+D
620 GOTO600
630 IFA(F+C,G+D)<>T THEN 670
640 A(F,G)=T
650 IFM=F AND N=G THEN 670
660 F=F-C:G=G-D:GOTO640
670 NEXTD:NEXTC
680 RETURN
690 GOSUB370
700 IF C>H THEN PRINT"SOY EL CAMPEON!"
710 IFH>C THEN PRINT"ERES EL MEJOR!"
720 IF H=C THEN PRINT "ES UN CLARO CASO DE EMPAT
E!"
730 END
740 CLS
750 X=ASC("X"):O=ASC("O"):REM LETRA O,NO CERO (O
)
760 DIM A(10,10)
770 FORB=1 TO 10:FOR C=1 TO 10
780 IFB<>1 AND C<>1 AND B<>10 AND C<>10 THEN A(B
,C)=ASC(".")
```

```
790 NEXTC:NEXTB
800 A(5,5)=X:A(6,6)=X
810 A(6,5)=O:A(5,6)=O
820 P=O
830 PRINT"Quieres mover tu primero?"
840 PRINT "S / N"
850 INPUT W$
860 GOSUB 370
870 IF W$="S" OR W$="s" THEN GOTO 260
880 GOTO30
```


A B C D E F G H

AVENTURA

Baja, lector, a la profundidad de las tinieblas. Armado tan sólo con tu ordenador y tu inteligencia has decidido enfrentarte a las fuerzas del mal.

Puedes encontrarte en una isla desierta, en una casa encantada, en un profundo calabozo cerca de las entrañas de la tierra o atrapado en un sistema de cavernas de un planeta perdido «en algún lugar de una galaxia lejana».

Los juegos de AVENTURA tienen lugar en todos estos escenarios y muchos más. Ven con nosotros y descubre la emoción de los mundos de la AVENTURA.

* * *

Se utiliza el término AVENTURA para describir una clase de juegos de ordenador en los que el jugador humano se mueve a través de una realidad alternativa. En este «otro mundo» hay monstruos que vencer, tesoros que descubrir, mapas que realizar y enigmas que resolver.

Para mucha gente los juegos de AVENTURA son los más emocionantes de todos los juegos que se pueden jugar con el ordenador. A quienes les gusten los juegos que exigen la adopción de un papel, estos juegos de aventura les resultarán todavía más divertidos.

En este libro he incluido tres juegos de AVENTURA. Dos de ellos ofrecen una dificultad media y el tercero —un programa inmenso que ocupa más de 16K— le hará rascarse la cabeza a más de uno intentando descubrir qué es lo que está pasando en el programa.

Una característica de los verdaderos juegos de AVENTURA es que la realidad que construyen es una realidad *consistente*. Es decir, el mundo creado en el programa de aventura es sólido y, salvo ciertos acontecimientos específicos a cada juego, tales como un terremoto, o un encantamiento mágico, las partes de ese mundo no cambian al azar. Si atravesando cierto bosque se pasa ante «un retorcido olmo en cuyo nudoso tronco están grabadas las iniciales de su mejor amigo», el mismo olmo seguirá estando allí cuando uno vuelva a pasar. En una AVENTURA construida como es debido, los ríos permanecen en su lugar, los muros de la torre no se mueven misteriosamente y cambian de lugar cada vez que uno se vuelve de espaldas y los objetos que se depositan en una caverna dentro de un laberinto subterráneo, no aparecen repentinamente cien cavernas más allá (a no ser que hayan sido bendecidos con alguna forma mágica de autotransporte).

La realización de mapas es una de las habilidades del verdadero jugador de AVENTURAS para quien es además un deleite. Abrirse camino en un medio imaginario (pero consistente), luchar con monstruos y encontrar tesoros, descubrir misterios al tiempo que uno sube y baja escaleras, salta, explora túneles, se pierde en laberintos y otras cosas por el estilo sólo es fascinante si el mundo que se está explorando puede traspasarse a un mapa. Por ejemplo, el jugador debería estar preparado para esperar que los dos extremos de un túnel aparezcan siempre en los mismos lugares, de modo que cuando más tarde, en otra partida y desde una posición diferente, vuelva a pasar ante una entrada de ese túnel, tendrá una nueva clave para incluir en su mapa del entorno. Todos los «mundos» creados en los juegos incluidos en este capítulo pueden ser traspasados a un mapa y los dos primeros incluso ofrecen en ocasiones una visión de la escena desde arriba. El tercero no lo hace y es precisamente éste el último que debe atacar el lector. Sólo cuando esté seguro de que se puede mover como quiera a través de los dos mundos más simples, podrá meterse con el tercero.

La adopción de un papel es una parte de la vida. No cabe ninguna duda de que en nuestra actividad cotidiana se dan no pocas situaciones en las que, hasta cierto punto, nos vemos obligados a adoptar un papel. En la mayoría de los trabajos esperan, por muy lamentable que sea esto, que uno se comporte como si las necesidades e intereses de la empresa fueran los suyos propios; como si el entusiasmo que uno demuestra un día tras otro ante clientes o compañeros de trabajo fuera un entusiasmo real, salido de lo más profundo de uno mismo. Es posible que uno encuentre que ante nuevas situaciones puede apren-

der a actuar con dominio ocultando así su inseguridad; pero lo más curioso es descubrir que si se tiene cierta habilidad, la gente toma lo que no es más que un papel por la verdadera personalidad de uno.

La adopción de un papel que tiene lugar en el medio imaginario del juego de AVENTURA es mucho más interesante; generalmente, el papel que se adopta en éste suele estar mucho más lejos del yo real y cotidiano de cada uno. En estos términos, hasta un juego como «Policías y ladrones» puede convertirse en un ejercicio fascinante de adopción de un papel.

Así pues, todos hemos adoptado un papel alguna vez, aun cuando esto no nos resultara tan divertido como formar parte en una campaña para rescatar a un bello príncipe encarcelado en una torre. Las AVENTURAS que se ofrecen en este libro no son en modo alguno tan elaboradas como algunas de las que se ofrecen a la venta, pero, sin embargo, son más consistentes y están mucho mejor organizadas que la mayoría de las que yo he podido ver y además tienen la ventaja de que le ofrecen al lector una estructura, un marco en el que poder desarrollar sus propios programas de AVENTURA.

Los juegos que exigen la adopción de un papel y en los que además participa más gente, no sólo un ordenador y un jugador humano, del tipo del «Mazmorras y dragones» de TRS, suelen desarrollarse, según va avanzando el juego, de un modo bastante flexible. Bajo el control de un buen guardián de mazmorra, un juego que exija la adopción de un papel por parte de los jugadores puede evolucionar en direcciones que los jugadores ni siquiera habían imaginado cuando fijaron el escenario inicial. Lamentablemente, los juegos de ordenador en los que el jugador adopta un papel no pueden ser tan flexibles. En muchos aspectos cuando se juega a un juego de aventuras con el ordenador, lo que el jugador hace es intentar resolver el enigma, o la serie de ellos, que ha creado la persona que hizo ese programa. Puede usarse el ordenador como una ayuda para fijar el escenario inicial y determinar dónde estarán localizados los objetos o, quizá, a dónde llevará una determinada escalera, pero es el programador al escribir el programa quien fija la forma general del mundo o del medio en que tiene lugar ese juego.

En un programa como LA FORTALEZA DE LOS ENANOS se puede ver esto en acción. En este programa el jugador tiene que lograr salir de un laberinto para descubrir el lugar en que se oculta el tesoro de los enanos. Es cierto que el ordenador esconde el tesoro y que modifica, hasta cierto punto, el «laberinto 3-D», pero eso es todo.

Por supuesto, el ordenador se utiliza también para producir información para el jugador cuando el juego está ya en marcha y su labor en este punto es muy valiosa.

Hubo una época en la que las guerras o batallas más conocidas se repetían una y otra vez, pero no en el verdadero campo de batalla, sino sobre la mesa. Las batallas napoleónicas se perdían y ganaban sobre un mapa en el que no faltaban los ríos y los bosques y sobre los que los jugadores repetían las principales contiendas o se inventaban unas nuevas. Posteriormente, los aficionados pensaron que estos juegos de la guerra podían ser todavía más divertidos si se le prestaba tanta o más atención a los combatientes individuales que a los pelotones de soldados. Esta atención en la identidad individual llevó directamente a los juegos que exigen la adopción de un papel que conocemos hoy, ya que los jugadores se dieron cuenta de cuán gratificador era adoptar el papel de un determinado líder.

Los juegos de «Mazmorras y dragones» son el desarrollo directo de aquellos juegos de guerra en los que los jugadores adoptaban la identidad de las personas concretas que habían tomado parte en cada batalla. Dos de estos jugadores, Dave Arnson y E. Gary Gygax, decidieron codificar una realidad totalmente separada que inicialmente sacaron a la luz en 1974 en forma de una serie compuesta de tres folletos a la que le dieron el título de *Dungeons and Dragons* (Mazmorras y dragones). El juego tuvo un éxito enorme y hoy se pueden contar por cientos de miles las personas que dedican una parte significativa de su tiempo libre a convertirse en duendes, magos, brujos o clérigos que se adentran en las profundidades de la tierra para explorar, luchar, conquistar e incluso, a veces, morir, en su camino.

A partir de la idea original de «Dungeons and Dragons» surgieron toda una multitud de otros juegos, como el «Runequest» (Chaosium Ltd., Albany, California), el «Bushido» y «Chivlary and Sorcery» (Fantasy Games Unlimited, Roslyn, New York), «Tunnels and Trolls» (Flying Buffalo, Inc., Scottsdale, Arizona), «Traveller» (Game Designers' Workshop, Bloomington, Illinois) y «Heroes of Olympus» (Task Force Games, Amarillo, Texas).

Will Crowther y Don Woods entraron a formar parte de la historia de la informática en 1976 cuando introdujeron el primer juego con adopción de papel —simplemente llamado AVENTURA— en un ordenador de la Universidad de Stanford. El programa lo había escrito, en realidad, Crowther, siendo elaborado posteriormente por Woods, pero hoy se les reconoce a los dos como los creadores de éste. Aunque

el juego se puso rápidamente de moda en toda América, las compuertas no se abrirían hasta que el microordenador pusiera el poder de la máquina al alcance de millones de personas dando lugar a una verdadera inundación de este tipo de juegos. Una memoria más barata facilitaba la creación de aventuras extremadamente elaboradas, de modo que hoy, el aficionado a los juegos de aventuras informatizados no puede quejarse de falta de programas, pues es uno de los campos mejor surtidos.

Posiblemente Crowther y Woods no fueron del todo conscientes de que lo que habían creado era mucho mejor de lo que ellos suponían ya que incluso hoy, a pesar de la multitud de programas existentes en el mercado, el programa AVENTURA original sigue siendo la estrella. Actualmente son por lo menos cinco las compañías que venden, sólo para el Apple, versiones diferentes del AVENTURA original.

Este programa original se escribió en Fortran y cuando pasó a formar parte de la biblioteca de libre intercambio de programas organizada por la Digital Equipment Computers Users' Society (la mayoría de cuyos usuarios utilizaban los sistemas Dec's DCP-11), no pasó mucho tiempo antes de que llegara a ser el programa más utilizado. Dos años más tarde, otro programa de aventuras, Dungeon, pasó a formar parte de Decus. Este programa llevaba la idea de AVENTURA mucho más lejos que el de Crowther y sus autores eran Tim Anderson, Marc Blank, Bruce Daniels y Dave Lebling, todos ellos pertenecientes a la Programming Technology Division del MIT Laboratory for Computer Science.

El libro de Addison y Wesley *Apple Computer Software* (editado por Jeffrey Stanton y John Dickey, The Book Co., Lawndale, California, 1981) dedica 29 páginas a la descripción de juegos de fantasía, adopción de papel por parte del jugador, aventuras y guerra existentes para el Apple II. Si esta marca de ordenador está tan bien respaldada en este campo de la programación, lo mismo podemos pensar de otras conocidas marcas de ordenadores.

Pero con todo y con eso, independientemente de todos los programas de aventuras que se pueden comprar para ejecutar en los diferentes ordenadores, la emoción de crear programas propios también en este campo, el más peculiar quizá en lo que se refiere a la programación de juegos, es una experiencia digna de tener en cuenta. Los programas incluidos en este capítulo le ofrecerán al lector, sin lugar a dudas, el marco apropiado para desarrollar los suyos propios. Lo que

yo le sugiero al lector es que los ejecute primero y después los elabore a su gusto (o al menos hasta que se le agote la memoria).

Aunque parezca que las aventuras basadas en gráficos podrían llegar a desplazar a las basadas en un texto, la atracción que ejercen estas últimas no ha disminuido en absoluto. Así como la televisión no convirtió a la radio en algo obsoleto —porque la imaginación es mucho más fuerte que cualquier cosa que el hombre pueda crear— del mismo modo las aventuras gráficas no han quitado ningún valor a los juegos basados en un texto en los que la mente humana se dispara y produce mundos mucho más fantásticos que los que pudiera producir cualquier inteligente programador con la ayuda de los sofisticados gráficos existentes en las máquinas de hoy.

Sin embargo, es probable que haya quien desee hacer la presentación de la salida del texto de estos juegos más atractiva añadiéndoles color y sonido cuando vea que está preparado para hacerlo y utilizando las características peculiares de su máquina para que la diversión a la hora de jugar con ella a uno de estos juegos sea total.

De las compañías que se dedican a la producción de programas, una de las más importantes a nivel internacional es Adventure International, que fue fundada por Scott Adams. En las exposiciones y ferias de la informática las exhibiciones que presenta esta compañía suelen ser las más imaginativas de las realizadas en el campo de la programación de juegos y, además, es bueno saber que sus programas son tan interesantes como la misma exhibición. No deje de experimentar el placer de ejecutar un programa de Scott Adams a su debido tiempo y verá lo lejos que se ha llegado en este género de programas.

Conocí a Scott en Atlanta, en una feria de la informática y le encontré agradable, tranquilo y reservado. En cierto modo esperaba que se pareciera a uno de esos hechiceros que habitan sus mazmorras o a un dragón de los que aparecen en sus sistemas de cavernas. Hubo algo, sin embargo, que sobrepasó todo lo que yo hubiera podido esperar. Era el día Halloween y Scott y su equipo aparecieron todos perfectamente disfrazados como los personajes de uno de sus programas. La misma visión de la vida que le hace decir a los miembros de su equipo que se disfracen para ir a trabajar, permeabiliza todos sus programas. Empiece con los programas que aquí se incluyen y que son mucho más simples que los de Scott y después vaya intentándolo de un modo gradual con los comerciales. Posiblemente el programa de Scott Adams Adventureland sería el mejor para empezar con éstos, antes de pasar a otros del mismo Adams o de otras compañías.

El programa de Avalon Hill, Empire of the Over-Mind, es otro de los clásicos en este campo y merece la pena echarle un vistazo. La lista podría hacerse interminable ya que cada vez son más los programas que se pueden añadir. En las revistas de informática o los clubs de programadores encontrará información relativa a los programas que van apareciendo en este campo.

* * *

Creo que ya ha llegado el momento de pasar a los tres programas que componen este capítulo. Si quiere profundizar en este campo, puede consultar la amplia bibliografía existente, de la que además puede servirse a la hora de escribir sus propios programas. En esta bibliografía se incluye:

— «Putting Adventure in Adventure Games», artículo en *Creative Computing*. Agosto 1981, por Robert Plamondon.
— «Graphic Adventures on the Atari», artículo en *Creative Computing*. Agosto 1982, por John Anderson.
— «Fantasy Games», artículos en *Creative Computing*. Marzo y mayo 1981, por David Lubar.
— «An Adventure in small computer Game Simulation», artículo en *Creative Computing*. Agosto 1979, por Scott Adams.
— «Kings of the Castle Walls», artículo en *Practical Computing*. Marzo 1982, por Dennis Ellis.
— «Adventure Writing», boletín de 16 páginas editado por: *Aardvark-80,* 2352 S. Commerce Walled Lake, MI 48088, USA.
— «Fantasy Role-Playing Games», Holmes y Eric. *Hippocrene Books Inc.,* New York, 1981.
— «What is Dungeons and Dragons?». Butterfield, Parker y Honigman. *Penguin Books*. New York, 1982.
— «Dicing with dragons. An introduction to role-playing games». Livingstone, Routledge y Kegan Paul, London. Melbourne & Henley.

LA FORTALEZA DE LOS ENANOS

Desciende, amigo lector, a las entrañas de la tierra donde se encuentra el reino de los enanos. El peligro se cierne a tu alrededor a medida que bajas, pero tu codicia te empuja hacia adelante. Cierto día, investigando entre los polvorientos montones de manuscritos sin catalogar de la sala 546B del Museo Británico, cayó en tus manos un mapa, casi ilegible, que conducía al oro atesorado por los enanos y desde ese día la idea de encontrarlo no ha dejado de obsesionarte.

Según te vas adentrando en el laberinto te das cuenta de que los enanos que ocultaron el oro allí 7389 años atrás, han dejado de existir hace miles de años. Así pues, no es a los guardianes de la fortaleza a quienes has de temer, sino que el principal peligro lo supone el trazado mismo del sistema de cavernas que has de atravesar antes de acariciar el oro.

En la FORTALEZA DE LOS ENANOS el jugador se encuentra en la caverna que guarda el oro. Cada vez que se juega, el oro puede estar en uno de tres lugares diferentes. La única información con que cuenta el jugador en su avance a través de la caverna es la que le proporciona el Rayo Oráculo de los Enanos, ante el que pasa el inicio de su incursión en el sistema. Este le proporciona información sobre la localización del oro, pero el jugador ha de aprender a interpretar esa información antes de poder hacer uso de ella.

Otro tipo de información se le ofrece en la forma de una clara exposición relativa a las direcciones hacia las que se puede avanzar

desde la posición que se ocupa en ese momento. Después de cada movimiento o jugada («paso») en la pantalla aparecerá una información similar a esta:

```
MOVIMIENTO NUMERO  1

NORTE: MURO
SUR: MURO
ESTE: ABIERTO
OESTE: MURO

EL RAYO ORACULO DE LOS ENANOS INDICA  1320
```

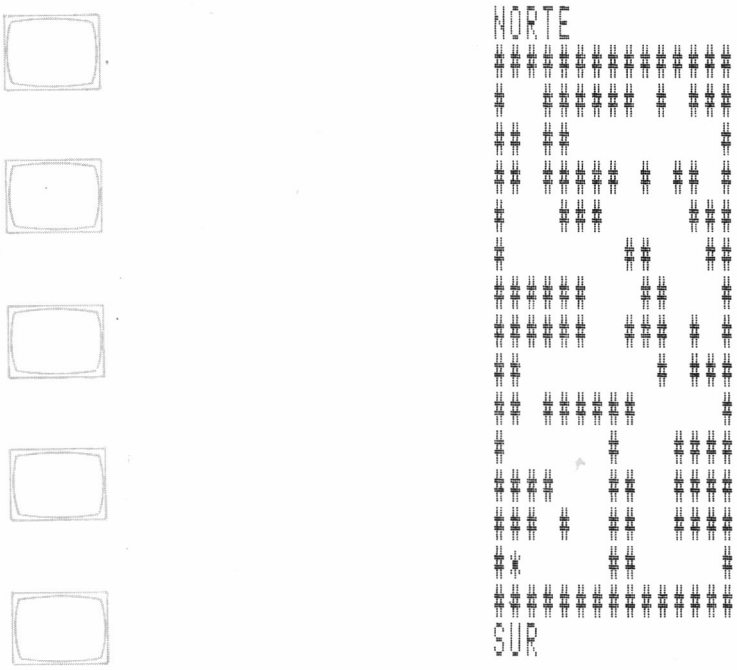
Acto seguido se le pedirá que introduzca la dirección hacia la que desea avanzar; como sigue:

```
EN QUE DIRECCION QUIERES
MOVERTE?

N - Norte, S - Sur
E - Este, O - Oeste, A - AYUDA
?
```

No se le permitirá, como probablemente se haya dado cuenta sin necesidad de decírselo, atravesar los muros.

No es necesario decir que el objetivo del juego es llegar al tesoro lo más rápidamente posible. Para yudarle a lograr su objetivo, hay un mapa del sistema visto desde arriba; mapa que puede llamar de vez en

cuando. El aspecto del mapa es el siguiente (los muros están marcados en # y el jugador es *):



Hay dos trampas que merece la pena tener en cuenta antes de llamar para que aparezca el mapa. En primer lugar, éste se limita a indicar la posición del jugador sin decir nada de la localización del tesoro. En segundo lugar, cada vez que introduce la clave "A" para echarle un vistazo al mapa, se le penaliza con 15 pasos. Conque le interesa utilizar esta opción lo menos posible.

Aunque hay algunas características comunes a todas las cavernas generadas por este programa (dictadas por las instrucciones data de las líneas 870 a 960), la caverna varía cada vez que se ejecuta el programa. Asimismo, como ya dije con antelación, el oro puede encontrarse en tres lugares diferentes dentro del sistema de cavernas.

La salida del programa, tal como se ofrece en términos de las direcciones que uno puede tomar desde la posición en la que se encuentra, puede modificarse para producir una salida "laberinto 3-D".

El jugador sabe, de una jugada a otra, en que dirección va y el ordenador sabe qué direcciones desde esa posición representan posibles jugadas o pasos. Es necesario añadir una subrutina que dibuje la situación inmediatamente, escogiendo entre una serie de imágenes (una con la zona que se extiende ante uno libre, pero con sólidos muros a los dos lados; otra con entradas hacia la derecha y hacia la izquierda, pero con el paso hacia adelante bloqueado) para producir una imagen continua de la escena. Esto puede ser muy efectivo, y efectista, y quizá merezca la pena proponérselo como ejercicio de programación.

Y finalmente aquí está el listado de LA FORTALEZA DE LOS ENANOS de modo que ya puede lanzarse a la aventura en busca del oro:

```
   10 REM La fortaleza de los enanos
   20 GOSUB 640
   30 GOSUB 480
   40 REM ****************************************
*****************
   50 M=M+1
   60 CLS:PRINT:PRINT
   70 PRINT"MOVIMIENTO NUMERO "M
   80 PRINT
   90 PRINT "NORTE: ";
  100 IFA(D+1,E)=S THEN PRINT "ABIERTO"
  110 IFA(D+1,E)=X THEN PRINT "MURO"
  120 PRINT "SUR: ";
  130 IFA(D-1,E)=S THEN PRINT "ABIERTO"
  140 IFA(D-1,E)=X THEN PRINT "MURO"
  150 PRINT "ESTE: ";
  160 IFA(D,E+1)=S THEN PRINT "ABIERTO"
  170 IFA(D,E+1)=X THEN PRINT "MURO"
  180 PRINT "OESTE: ";
  190 IFA(D,E-1)=S THEN PRINT"ABIERTO"
  200 IFA(D,E-1)=X THEN PRINT "MURO"
  210 PRINT
  220 PRINT "EL RAYO ORACULO DE LOS ENANOS INDICA
";
  230 PRINT 100*ABS(Z-D)+10*ABS(Y-E)
  240 REM ****************************************
*****************
```

```
250 PRINT:PRINT:PRINT
260 PRINT"EN QUE DIRECCION QUIERES"
270 PRINT"MOVERTE?"
280 PRINT:PRINT"N - Norte, S - Sur"
290 PRINT"E - Este, O - Oeste, A - AYUDA"
300 INPUTA$
310 IFA$="N" AND A(D+1,E)=X THEN 300
320 IFA$="S" AND A(D-1,E)=X THEN 300
330 IFA$="E" AND A(D,E+1)=X THEN 300
340 IFA$="O" AND A(D,E-1)=X THEN 300
350 IF A$="A" THEN GOSUB 480
360 IF A$="N" THEN D=D+1
370 IF A$="S" THEN D=D-1
380 IF A$="E" THEN E=E+1
390 IF A$="O" THEN E=E-1
400 IFZ=D AND Y=E THEN 430
410 GOTO50
420 REM ****************************************
*************
430 PRIN
440 PRINT" EN TAN SOLO "M" MOVIMIENTOS!"
450 GOSUB 500
460 END
470 REM ****************************************
*************
480 REM auxilio
490 CLS
500 PRINT
510 PRINT"NORTE"
520 FORB=15 TO1 STEP-1
530 FORC=1 TO 15
540 IFA(B,C)=X THEN PRINT"#";
550 IFB=D AND C=E THEN PRINT"*";:GOTO570
560 IFA(B,C)=S THEN PRINT" ";
570 NEXTC:PRINT:NEXTB
580 PRINT"SUR"
590 M=M+15
600 FORJ=1 TO 2000:NEXTJ
610 CLS:A(D,E)=S
620 RETURN
630 REM ****************************************
*************
640 CLS
```

```
650 N=1:PRINT"APRIETA CUALQUIER TECLA"
660 N=N+1:IF INKEY$=""THEN 660
670 CLS:PRINT"ESPERA UN MOMENTO...POR FAVOR"
680 RANDOMIZE
690 DIMA(15,15)
700 B=INT(RND(-N)*3)+1
710 Z=14:Y=14
720 IFB=2 THEN Y=2
730 IFB=3 THEN Z=2
740 X=1:S=2
750 FORB=1 TO 15:FORC=1 TO 15
760 A(B,C)=X:IFRND(-N)>.8 THEN A(B,C)=S
770 IFC<2 OR C>14 ORB<2 ORB>14 THEN A(B,C)= X
780 NEXT C:NEXT B
790 D=2:E=2
800 FOR F=1 TO 68
810 READ B:READ C
820 A(B,C)=S
830 NEXT F
840 M=-15
850 RETURN
860 REM ****************************************
***************
870 DATA 2,2,2,3,2,4,2,5,2,6,2,7
880 DATA 3,7,4,7,5,7,5,6,5,5,5,4,5,3,6,3
890 DATA 7,3,7,4,7,5,7,6,7,7,7,8,7,9,9,8
900 DATA 9,9,10,8,10,7,10,6,10,5,10,4,8,8
910 DATA 10,3,11,3,12,3,13,3,14,3,14,2,7,10
920 DATA 6,10,5,10,4,10,3,10,2,10,2,11,2,12
930 DATA 2,13,2,14,6,11,6,12,6,13,6,14,7,12
940 DATA 14,12,8,12,8,14,9,12,9,13,9,14,10,12
950 DATA 11,9,11,10,11,11,11,12,12,9,13,9,13,10
960 DATA 13,11,13,12,13,13,13,14,14,14
```

EL DUQUE DE DRAGONFEAR

En este juego se introduce al jugador en otro intrincado mundo de cavernas subterráneas, pero a diferencia del anterior, éste está habitado por toda una extensa gama de pavorosas posibilidades como pozos de arenas movedizas y dragones y por otras no tan pavorosas, como cuevas mágicas que pueden transportarle a uno a la aventura de un lugar a otro del País de los Dragones, y cuevas repletas de oro.

Nada más empezar a ejecutar el programa, se le ofrece al jugador la siguiente información:

```
Bienvenido al mundo terrorifico de los dragones

Tu mision es explorar las
Cavernas de este mundo, buscando
Tesoros, e intentando evitar los
Terribles dragones que pueblan estas
Cavernas.    Como te llamas ?

     De acuerdo, Duque JAVIER

    Empiezas esta exploracion
    Con 25 unidades de carisma, y
    Debes terminar la tarea antes
    De que se agote el carisma. Pierdes
    Una unidad por cada movimiento que hagas

    APRIETA CUALQUIER TECLA PARA EMPEZAR
```

Una vez asimilado esto, el jugador puede echar una rápida ojeada al mapa del País de los Dragones:

```
X X X X X X X X X X
X . . . . D ? . . X
X . . . . . X $ . X
X . . . X . ? . D X
X . . . Q Q . ? . X
X $ ? . H . . $ . X
X . X . . D . D . X
X . $ Q . . . . . X
X . ? Q X . X . . X
X X X X X X X X X X
```

En este mapa, la X representa un muro que no se puede atravesar o una cueva a la que no se puede acceder; H (de «humano») representa al jugador que inicia su aventura más o menos en el centro del sistema; $ son las cuevas del tesoro; ? representa las cuevas en las que el jugador es mágicamente transportado y Q le indica qué cuevas contienen arenas movedizas. Está claro que en este mundo abundan tanto los placeres como los peligros potenciales.

Los que sean ya veteranos en esto de los juegos de ordenador, se darán cuenta en seguida de que este juego, EL DUQUE DE DRAGONFEAR, es el desarrollo de los juegos denominados Hunt the Wumpus. Wumpus es uno de los grandes y más antiguos estándares de la historia de los juegos de ordenador en los que el jugador se encuentra en un sistema de cavernas a la búsqueda de uno o varios animales míticos conocidos en singular como Wumpus (y éstos suelen ser animales bastante singulares); el plural es Wumpii, o al menos eso dicen quienes se preocupan por esas cosas. Sea como fuere, en este tipo de juegos lo que se hace esencialmente es fijar una especie de red o cuadrícula y los objetos, personas, efectos, monstruos o lo que quiera que se escoja, están situados en diversos lugares de esta cuadrícula. De vez en cuando, y siempre que el programador del juego lo decida, uno o varios de los contenidos de una cuadrícula determinada pueden moverse a otras. Por supuesto, el jugador humano también puede moverse de una cuadrícula a otra.

Generalmente, no existe un mapa visible de todo el sistema, aunque algunos programas (como EL DUQUE DE DRAGONFEAR) sí que proporcionan este tipo de información al jugador. El primer programa Wumpus apareció en 1975 y se debe a Gregory Yob. En el maravilloso libro de David Ahl *More BASIC Computer Games* (Creative Computing Press, Morristown, New Jersey, 1979), el señor Yob explica que en cierta ocasión que se encontraba de visita en el People's Computer Company, en Menlo Park, California, para ver los programas que estaban desarrollando, vió que se estaban escribiendo en ese momento tres programas diferentes del tipo de «caza algo» cada uno de ellos en una red de 10 × 10 y en cada uno de los cuales se le proporcionaba información al jugador en términos de «no, no... inténtalo hacia el noroeste».

Desencantado, el señor Yob volvió a la triste terminal que compartía (eran días aquellos, mi querido lector, en los que los seres humanos no tenían pequeños ordenadores a su disposición) y no dejó de martillearlo hasta que consiguió el programa que con el tiempo se convertiría en el primer «Hunt the Wumpus». El señor Yob hizo que la criatura habitara un sistema de cavernas de forma dodecaédrica.

El genio inventivo del señor Yob no se detuvo aquí. Posteriormente pobló el sistema con «animales mágicos» que transportaban al jugador al azar de un lugar a otro dentro del sistema, con pozos sin fondo y se aseguró de que las flechas con que se equipa al jugador antes de iniciar su aventura pudieran girar en su vuelo.

El lector podrá darse cuenta ahora de cómo esta estructura se convirtió en el corazón de los programas Wumpus que han proliferado por todo el mundo y de que, en efecto, constituye el armazón básico sobre el que está construido EL DUQUE DE DRAGONFEAR. Este programa le ofrece más control sobre la salida del programa, añade tesoros a algunas de las cavernas e impone un límite de tiempo al juego.

Asimismo, el jugador está equipado con un «amuleto mágico» que puede ver lo que hay en las cavernas que le rodean. Lamentablemente, como el jugador no es una persona experta en esto de los encantamientos mágicos, el viejo amuleto no funciona todo lo bien que podría. Sólo le dice el contenido de una de las ocho cavernas que rodean a la caverna en la que se encuentra y, lo que es peor, no le dice a cuál de ellas se está refiriendo. En realidad, no está tan mal como parece y el jugador se quedará sorprendido al ver todo el uso que puede dar a la limitada información que le proporciona el amuleto.

Además de esto, al inicio del programa se le ofrece una rapidísima visión del sistema de cavernas tomado desde arriba; esta visión se repite alguna que otra vez a medida que avanza el juego.

Como puede ver en la muestra de ejecución que acompaña a esta introducción, entre las opciones que se le presentan antes de cada jugada, hay una opción "L" para luchar. Al presionar "L" el programa se dirigirá a la subrutina que comienza en la línea 930 que verifica si todavía le quedan flechas y si comprueba que le quedan, le permite escoger la dirección en que quiere lanzarla partiendo de la posición que ocupe en ese momento. Si logra matar algún animal conseguirá una buena recompensa monetaria.

A continuación le ofrecemos algunas jugadas del programa en acción:

```
Duque JAVIER, estais en la cueva 55
Tu Amuleto nos muestra
que hay
 Un dragon
 en las proximidades

Os quedan aun  23  unidades de Carisma

Que quereis hacer?
N - norte, S - sur
E - este,  O - oeste
L - luchar contra el Dragon
T - terminar el juego
                        Horror...arenas movedizas!
                         Horror...arenas movedizas!
?N                        Horror...arenas movedizas!
                           Horror...arenas movedizas!
                            Horror...arenas movedizas!
                             Horror...arenas movedizas!
                              Horror...arenas movedizas!
                               Horror...arenas movedizas!
```

```
Duque JAVIER, estais en la cueva 45
Tu Amuleto nos muestra
que hay
 Un dragon
 en las proximidades

Os quedan aun  24  unidades de Carisma

Que quereis hacer?
N - norte, S - sur
E - este,  O - oeste
L - luchar contra el Dragon
T - terminar el juego

?N
Habeis ido a parar al nido de un
dragon...preparad vuestras oraciones
Se esta despertando... y te ha visto
Y ahora se dispone a atacar...


Horror...arenas movedizas!
 Horror...arenas movedizas!
  Horror...arenas movedizas!
   Horror...arenas movedizas!
    Horror...arenas movedizas!
Todo tu carisma esta
....agotado....
```

```
X X X X X X X X X X
X . . . . D . . X
X . X . . Q . . X
X . H . . . ? . $ X
X ? D . . . Q ? . X
X . Q X . . . . . X
X X Q . $ X . . . X
X . . . X $ . ? $ X
X . D ? . $ . . . X
X X X X X X X X X X
```

Y ahora, el listado que le permitirá convertirse en el Duque de Dragonfear:

```
   10  REM El duque de Dragonfear
   20 GOSUB 1280
   30 CLS:PRINT:PRINT:PRINT
   40 GOSUB1200
   50 Q=INT(RND(-N)*7)
   60 IFQ=0 AND E<>55 THEN GOSUB 1200
   70 CLS:PRINT:PRINT:PRINT"Duque "A$", estais en
la cueva"E
   80 IF G>0 THEN PRINT "Llevais oro por valor de
$"G
   90 GOSUB760
  100 PRINT:PRINT"Os quedan aun "25-H" unidades de
 Carisma"
  110 PRINT:PRINT"Que quereis hacer?"
  120 PRINT"N - norte, S - sur"
  130 PRINT"E - este,  O - oeste"
  140 PRINT"L - luchar contra el Dragon":PRINT"T -
 terminar el juego":PRINT
  150 INPUT Z$:U=0:IFZ$<>"N"ANDZ$<>"S"ANDZ$<>"E"AN
DZ$<>"O"ANDZ$<>"L"ANDZ$<>"T"THEN 150
  160 IFZ$="N" ANDA(E-10)=88 OR Z$="S" AND A(E+10)
=88 OR Z$="E" ANDA(E+1)=88 OR Z$="O" AND A(E-1)=88
THEN PRINT "No puedes ir en esa direccion!!":GOTO1
50
  170 IFZ$="T"THENQ=9 :GOTO1170
  180 A(E)=46:IFZ$="N" THEN E=E-10
  190 IFZ$="S"THENE=E+10
  200 IFZ$="E"THENE=E+1
  210 IFZ$="O"THENE=E-1
  220 IFZ$="L"THENGOSUB930
  230 IFA(E)=63 THEN GOSUB 310:REM Magia
  240 IFA(E)=68 THEN GOSUB 400:REM Dragon
  250 IFA(E)=81 THEN GOSUB 550:REM Arenas movediza
s
  260 IFA(E)=36 THEN GOSUB 630:REM Oro
  270 H=H+1:IFH=25  THEN Q=9:GOTO1160
  280 GOSUB1610
  290 GOTO50
```

```
300 REM ****************************************
**************
310 REM MAGIA
320 PRINT:PRINT"Duque "A$", habeis caido"
330 PRINT" en una cueva magica, y ahora "
340 PRINT"sereis transportado a otra ..."
350 GOSUB1610
360 A(E)=46
370 E=INT(RND(-N)*76)+12:IFA(E)=88THEN 370
380 RETURN
390 REM ****************************************
**************
400 REM Dragon
410 PRINT"Habeis ido a parar al nido de un"
420 PRINT"dragon...preparad vuestras oraciones "
430 GOSUB1610
440 M=RND(-N):IFM<.2 THEN PRINT"Se ha ido voland
o":RETURN
450 PRINT"Se esta despertando... y te ha visto "
:GOSUB1610
460 GOSUB1610
470 IFM>.8499999 THEN PRINT"Pero ha comido hace
poco":PRINT"Asi que se vuelve a dormir":RETURN
480 PRINT"Y ahora se dispone a atacar..."
490 GOSUB1610
500 IFM>.95 THEN PRINT"Pero luchas... y le vence
s!!!":RETURN
510 PRINT"Adios, Duque "A$
520 GOSUB 1610
530 Q=9:GOTO1160
540 REM ****************************************
*********************
550 REM Arenas movedizas
560 FORJ=1TO12
570 FORK=1 TO J:PRINT" ";:NEXT
580 PRINT"Horror...arenas movedizas!"
590 FORO=1TO90:NEXT
600 NEXT
610 Q=9:H=0:GOTO1160
620 REM ****************************************
*****************
630 REM Tesoro
640 K=INT(RND(-N)*100)+100
```

```
650 FORJ=1TO12
660 FORZ=1TOJ:PRINT" ";:NEXT
670 PRINT"Un tesoro ! ! !"
680 FORO=1TO200:NEXT
690 NEXT
700 GOSUB1610
710 PRINT:PRINT"Has encontrado un cofre del"
720 PRINT"tesoro del dragon, valorado en $"K
730 G=G+K
740 RETURN
750 REM ****************************************
*******************
760 REM Detalles del Amuleto
770 Y=1
780 L=A(E+P(Y))
790 IFL<>46 THEN820
800 IFY<8 THEN Y=Y+1:GOTO780
810 IFL=46 THEN RETURN
820 PRINT"Tu Amuleto nos muestra"
830 PRINT"que hay"
840 IFL=88 THEN PRINT" Un muro solido"
850 IFL=63 THEN PRINT" Una cueva magica"
860 IFL=68 THEN PRINT" Un dragon"
870 IFL=81 THEN PRINT" Arenas movedizas"
880 IFL=36 THEN PRINT" Oro"
890 PRINT" en las proximidades"
900 GOSUB1610
910 RETURN
920 REM ****************************************
*******************
930 REM Ataque del dragon
940 PRINT
950 AR=AR-1:IFAR=0 THEN PRINT"Has usado todas":P
RINT"tus flechas....":GOSUB1610:RETURN
960 PRINT"Tienes "AR" flechas todavia"
970 SS=0
980 PRINT" en que direccion quieres"
990 INPUT"disparar (N,S,E,O)";S$
1000 IFS$="N" AND A(E-10)=68 THEN SS=1:YT=E-10
1010 IFS$="S" AND A(E+10)=68 THEN SS=1:YT=E+10
1020 IFS$="E" AND A(E+1)=68 THEN SS=1:YT=E+1
1030 IFS$="O" AND A(E-1)=68 THEN SS=1:YT=E-1
1040 PRINT
```

```
 1050 IFSS=0 THEN PRINT "No hay ningun dragon en e
sa direccion...":PRINT"Has malgastaso una flecha":
GOTO1150
 1060 PRINT"Bien hecho, Duque "A$
 1070 PRINT"Le has dado"
 1080 PRINT "a un feroz dragon"
 1090 GOSUB1610
 1100 IFRND(-N)>.3 THEN1140
 1110 PRINT"Le has matado!":A(YT)=46:K=INT(RND(-N)
*100)+100
 1120 PRINT
 1130 PRINT"Eres recompensado con $"K:G=G+K:GOTO11
50
 1140 PRINT"Pero solo le has herido..."
 1150 GOSUB 1610:RETURN
 1160 IFH<1 THEN PRINT" Todo tu carisma esta":PRIN
T" ....agotado....":GOSUB1610:GOTO1180
 1170 PRINT"Tienes" 25-H" unidades de carisma toda
via"
 1180 IFG>0 THEN PRINT"Has amasado una fortuna de
$"G"en oro"
 1190 PRINT:PRINT
 1200 A(E)=72
 1210 FORJ=1 TO 100
 1220 PRINT CHR$(A(J));" ";
 1230 IF 10*INT(J/10)=J THEN PRINT
 1240 NEXT
 1250 FOR Q=1 TO 1500:NEXT:GOSUB1610
 1260 IFQ=9 THEN END
 1270 RETURN
 1280 CLS:PRINT:PRINT:PRINT
 1290 PRINT"Bienvenido al mundo terrorifico de los
 dragones"
 1300 PRINT:PRINT"Tu mision es explorar las"
 1310 PRINT"Cavernas de este mundo, buscando"
 1320 PRINT"Tesoros, e intentando evitar los"
 1330 PRINT"Terribles dragones que pueblan estas"
 1340 PRINT"Cavernas.    Como te llamas ";
 1350 INPUT A$
 1360 CLS:PRINT:PRINT" De acuerdo, Duque "A$
 1370 PRINT:PRINT"Empiezas esta exploracion"
 1380 PRINT"Con 25 unidades de carisma, y"
 1390 PRINT"Debes terminar la tarea antes"
```

```
1400 PRINT"De que se agote el carisma. Pierdes"
1410 PRINT"Una unidad por cada movimiento que hag
as"
1420 N=1:PRINT:PRINT"APRIETA CUALQUIER TECLA PARA
EMPEZAR"
1430 N=N+1:IF INKEY$="" THEN 1430
1440 RANDOMIZE:CLS
1450 PRINT"Un momento, por favor, Duque "A$
1460 DIMA(100):H=0:Q=0:L=0:G=0:AR=6
1470 FORB=1 TO 100:A(B)=46
1480 IF B<12 OR B >90 OR 10*INT(B/10)=B OR 10*INT
(B/10)=B-1 THEN A(B)=88
1490 NEXT
1500 FOR B=1 TO 5:RESTORE:FORD=1 TO 5
1510 Z=INT(RND(-N)*76) + 12:IFA(Z)=88 THEN 1510
1520 READ C :A(Z)= C
1530 NEXT:NEXT
1540 DATA 88,63,68,81,36
1550 FORB=1 TO 8:READ Z:P(B)=Z:NEXT
1560 DATA-11,-10,-9,-1,1,9,10,11
1570 E=55
1580 RETURN
1590 REM PONGA EL TECLADO EN MAYUSCULAS
1600 REM ANTES DE COMENZAR A JUGAR
1610 FORO=1 TO  500:NEXT:RETURN
```

EL LEGADO BANNOCHBURN

Como indicaba al inicio de este capítulo, cuando uno escribe un programa de aventuras, lo que en realidad está haciendo (si lo está haciendo como es debido) es construir una realidad alternativa con sus propias reglas y consistencia interna. En un programa de aventuras que se atiene a las reglas, un objeto situado en una habitación permanecerá en ella hasta que el jugador lo mueva, los muros de la casa encantada que se está explorando no se moverán y cambiarán de lugar cada vez que uno les vuelva la espalda y los arroyos y los ríos no cambiarán su curso cada vez que el jugador decida atravesarlos.

El siguiente programa y el último de este capítulo, EL LEGADO DE BANNOCHBURN, está mucho más cerca de los programas de aventura estándar que comentaba en la introducción a este capítulo. En este programa todas las actividades (excepto el principio) tienen lugar en el castillo de Bannochburn. El castillo contiene una cantidad no determinada de monstruos, entre los que se incluyen un «malhumorado hechicero» y «la terrorífica criatura» que respira fuego, que vagan de un lado al otro del castillo durante todo el juego esperando que se les presente la ocasión de caer sobre el jugador para partirlo por la mitad. El castillo contiene asimismo botellas de poción mágica, arcones con buenas y malas sorpresas, papiros con encantamientos mágicos escritos en ellos y muchas cosas más. Si dijera ahora todo lo que el jugador puede encontrar, el juego perdería interés.

No es muy probable que en una sola partida se encuentre con todo lo que contiene el castillo. Asimismo, los monstruos tienen diferentes poderes cada vez que el jugador se enfrenta a esta aventura (es decir, cada vez que se ejecuta el programa) y los arcones y cofre dorado,

junto con el resto de los contenidos mágicos, se encuentran en diferentes lugares. Aunque los monstruos andan vagando de una estancia a otra del castillo, no sucede lo mismo con el resto de las cosas que éste contiene: no pueden aparecer y desaparecer al azar. Una vez que el jugador ha abierto el arcón número dos, no lo volverá a ver otra vez durante esa partida. Si hiere a un monstruo en combate, y vuelve a encontrarse con él posteriormente en la misma partida, verá que éste sigue herido. Cosas de este tipo ayudan a formar la «realidad» del universo ficticio.

El jugador no cuenta con un mapa del castillo. Una gran parte del placer que producen los juegos de aventura radica en el hecho de tener que descubrir cómo se relacionan unas partes del «universo de la aventura» con las otras y la realización de mapas es una de las técnicas cuyo desarrollo más divertirá al jugador. El castillo de Bannochburn fue cuidadosamente traspasado a un mapa antes de escribir el programa con el fin de asegurarme de que las estancias estaban situadas de un modo coherente y para animar al jugador a que realice su propio mapa del castillo a medida que lo explora. Una vez que el jugador se haga con un mapa razonablemente trazado, lo más seguro es que quiera comprobarlo paseándose por todas las estancias del castillo.

El objetivo del juego es dar con la Laguna Negra que está situada bajo el castillo, pero a la que se accede por una puerta normal desde el interior de éste. La situación de la Laguna Negra puede cambiar de una partida a otra, pero no cambiará de lugar una vez que la partida está en marcha. Esto asegura que el interés del jugador se mantendrá durante toda la partida, incluso después de que se las haya arreglado para levantar el plano del castillo.

Al inicio del juego se le otorgan al jugador tres atributos:

Magia - Fuerza - Inteligencia

El jugador juega toda la partida en términos de estos tres atributos, escogiendo para combatir cada monstruo aquel en el que se sienta más fuerte o en el que ese monstruo en particular sea más débil.

Los atributos cambian y se desarrollan a medida que avanza el juego, aumentando cuando el jugador gana una batalla y disminuyendo cuando la pierde.

Nada más empezar a ejecutar el programa, se le ofrece al jugador la siguiente información:

```
JAVIER, tus atributos son:
   Magia: 4
  Fuerza: 2
Sabiduria: 5

Estas a la entrada de un
antiguo castillo de
impresionante aspecto.Estas en
el lado norte del castillo
y al mirar hacia el sur, hacia
la solida estructura, te
das cuenta de que el portal de entrada
esta abierto y sin guardia

Que quieres hacer ahora?
```

La última pregunta "¿Qué quieres hacer ahora?" aparece después de la mayoría de las descripciones de las estancias y el jugador ha de dar una de las siguientes respuestas:

— N (ir hacia el norte).
— S (ir hacia el sur, ésta es la respuesta que tendría que dar en el caso de la primera información que hemos visto unas líneas más arriba).
— E (ir hacia el este).
— O (ir hacia el oeste).
— L (luchar; ésta es una de las dos respuestas posibles que el jugador puede dar al verse enfrentado a un monstruo).
— H (huir; ésta es la otra respuesta posible y no siempre es aceptada; "No, you must stand and fight", le dirá el ordenador. "No, has de quedarte y luchar".
— T (me rindo; ésta se usa en el caso de que el jugador quiera dar por terminado el juego antes de llegar al final).

La única respuesta que necesita en el caso de preguntas del tipo "Quieres abrir un arcón?", es "S" (sí) o "N" (no).

En gran medida este programa se ejecuta solo y vigila muchas de las tareas que en un juego de adopción de papel en el que interviniera más gente vigilaría un guardián de mazmorra.

Una de estas tareas es la ejecución de las batallas o melés que tienen lugar durante el juego. Todas las luchas se solucionan de acuerdo con la fuerza relativa de los protagonistas de la batalla y basándose en el resultado del lanzamiento de un dado contrapesado a favor del más fuerte de los dos contrincantes en lucha. Así pues, el jugador humano deberá seleccionar siempre el atributo con el que aventaja al monstruo, con lo que aumentará sus posibilidades de ganar esa batalla concreta.

Habíamos dicho antes que el objetivo del juego era llegar a la Laguna Negra, pero a esto habría que añadirle «con vida». Una vez que ha alcanzado la Laguna Negra, el jugador ha de luchar tres veces contra el guardián de ésta, una con cada atributo, teniendo que acabar esta lucha en tres partes con un total de 10 puntos por lo menos. El dinero que el jugador va recogiendo en sus correrías por el castillo, puede utilizarlo, justo antes de la lucha final, para comprar más puntos para sus atributos.

No quisiera que el proceso de descubrimiento del juego por parte del jugador quedara menoscabado, por eso no daré mucha más información aquí. Todo lo que puedo hacer es asegurar al lector que el trazado del castillo es coherente y puede ser traspasado a un mapa y que merece la pena tomarse la molestia de intentar descubrirlo. Observe que este programa ocupa más de 16K, de modo que en el caso de algunos sistemas, lo más probable es que haya que acortar las sentencias PRINT, asegurándose al mismo tiempo de que se deja espacio de trabajo, así como espacio para la longitud del listado.

La historia de «Bannochburn Castle» es sencilla y trágica al mismo tiempo. En la famosa Guerra de los Clanes, que asoló el norte de Escocia durante los últimos años del siglo XV, el señor de Bannochburn contrató los servicios de Wee Sporran McMerlin, un mago que originalmente había estado al servicio del rey Jaime en la Corte de Edinburgo. Cuando el sangriento Clan de los McKinna unió sus fuerzas a las del no menos sangriento Clan de McClaren y juntos atacaron Bannochburn, el señor de éste pidió a McMerlin que hiciera un hechizo que protegiera el castillo para siempre.

Lamentablemente, la inteligencia de Wee Sporran no era tan brillante como su magia y, en vez de invocar un hechizo que cuidara de sus habitantes, lo que hizo fue conjurar el ataque para proteger el

castillo. El señor, sus servidores y partidarios fueron asesinados, pero el castillo permaneció intacto sobreviviendo hasta el día de hoy. La misión del jugador ahora es explorarlo e intentar rescatar los secretos y el oro que oculta matando para ello al Guardián de la Laguna Negra, de modo que los actuales miembros del Clan Bannochburn (que ahora viven en Glasgow en un pisito sobre una tintorería) puedan recuperar la herencia que les pertenece. El jugador sentirá —y posiblemente con razón— que éstos se han de sentir sumamente agradecidos con él si logra matar al guardián liberando así al castillo de los fantasmales habitantes que hace tantos años introdujera Wee Sporran dentro de sus muros con la idea de protegerlo.

Observe que una vez que ha entrado en el castillo, no hay modo de retroceder. No puede volver a salir por la entrada principal, de modo que no se lo puede pensar dos veces. Toda la información se introduce en mayúsculas, así que el jugador habrá de asegurarse de que su ordenador está funcionando con mayúsculas antes de ejecutar el programa o de lo contrario, el ordenador ignorará toda esa preciosa información.

Buena suerte, lector, y quiera Dios que el Guardián caiga derrotado antes tus grandes poderes.

Y aquí están unos cuantos ejemplos del programa en funcionamiento:

```
JAVIER, tus atributos son:
   Magia: 5
  Fuerza: 4
Sabiduria: 3

Has entrado a la cueva de
las Brujas, con una marmita hirviendo
en un fuego de llamas verdes
en el rincon suroeste.
Esta habitacion rezuma azufre
ardiente, y el eco de
antiguos conjuros. Puedes salir
hacia el norte, el sur,
o el este

Que quieres hacer ahora?E
```

Este es el Salon mas suntuoso
del castillo, el Gran Salon,
con un impresionante techo
labrado.Puedes salir por las puertas
dobles del norte o por
las del este, tras las cuales
llega un sonido de musica.
A traves de las ventanas del muro
oeste, puedes ver el Jardin
de los Contornos, y mas alla,
las ventanas de una habitacion con
muchas, muchas pinturas valiosas.
Esta habitacion contiene el mas
mortal enemigo de cuantos entran el castillo
Se trata de Wolvling, el Hombre-Lobo, con
una fuerza de 6
una sabiduria de 5
y una capacidad magica de 1

Que quieres hacer ahora? N

En frente tuyo hay un cofre
etiquetado con un gran # 1

Vas a abrirlo?

Un estrano humo surge
haciendo que te duermas y
apoderandose de tus poderes magicos
En la pared hay una pequena caja fuerte dorada
y a su lado hay una llave...
Quieres abrir la caja fuerte?

Oyes un coro de
voces angelicales.....

Eres curado y refrescado...

Estas en el Hall de entrada,
del que cuelgan ricos
tapices.Las puertas dan hacia
el este y el sur, y hay un
portal abierto hacia el oeste
Esta habitacion contiene el mas
mortal enemigo de cuantos entran el castillo
Se trata de Wolyling, el Hombre-Lobo, con
una fuerza de 2
una sabiduria de 4
y una capacidad magica de 5

Que quieres hacer ahora?L

Con que caracteristica lucharas?
Tu magia es 4 , fuerza 3 , y tu sabiduria es 6
?

Y este es el listado completo:

```
   10 REM El legado Bannochburn
   20 REM TODAS LAS CONTESTACIONES EN MAYUSCULAS
   30 GOSUB4480
   40 CLS
   50 IFM(7)=0 AND F(7)=0 AND S(7)=0 THEN PRINT"La
 aventura ha terminado.":PRINT"Has agotado todos t
us poderes":PRINT:PRINT"Luchastes bravamente y bie
n":PRINT"pero no pudistes conseguirlo.":PRINT:PRIN
T "Adios....":END
   60 PRINT N$", tus atributos son:"
   70 IFM(7)>0 THEN PRINT "   Magia:"M(7)
   80 IF F(7)>0 THEN PRINT "  Fuerza:"F(7)
   90 IFS(7)>0 THEN PRINT "Sabiduria:"S(7)
  100 IF DINERO > 0  THEN PRINT "Riqueza: $"DINERO
  110 GOSUB 4990:REM Pausa
  120 GOSUB 3420:REM Habitacion
  130 M=0:IF Z>1 THEN IF RND(-1)>.5 THEN GOSUB 274
0
  140 GOSUB4990
  150 GOSUB1900
  160 GOSUB 4990
  170 GOTO40
  180 REM ****************************************
***************************
  190 REM Resolucion de la pelea
  200 PELEA=INT(RND(-1)*6)+1
  210 VICTORIA=0
  220 IF(DIF<0 AND PELEA >  ABS(DIF)) OR (DIF>0 AN
D PELEA<= DIF) OR (DIF=0 AND PELEA< 4) THEN VICTOR
IA =1
  230 GOSUB 4990
  240 RETURN
  250 REM ****************************************
***************************
  260 REM Subrutina de los monstruos
  270 IF Q=0 THEN RETURN
  280 ON Q GOSUB 310,320,330,340,350
  290 GOSUB 4990
  300 RETURN
```

```
310 PRINT"Hay un mastin enfurecido en":PRINT"la
habitacion. Tiene un grado":PRINT "magico de "M(1)
:PRINT"Su fuerza es "F(1)"and":PRINT"su sabiduria
es "S(1):RETURN
320 PRINT"La habitacion contiene una hidra de 7
cabezas.Su aliento":PRINT" Es fuego.Su sabiduria e
s "S(2)", mientra que su":PRINT"fuerza es de "F(2)
:PRINT"y sus poderes magicos son "M(2):RETURN
330 PRINT"Horror! Has tropezado con la guarida d
e":PRINT"un maligno demonio.Puedes ver a simple vi
sta":PRINT"que su fuerza es "F(3):PRINT"su habilid
ad magica es "M(3):PRINT"y su sabiduria es "S(3):R
ETURN
340 PRINT"Has viajado por la oscuridad.":GOSUB 4
990:PRINT"Algo se despierta.OH! es Inmensus, el qu
ebrantahuesos":PRINT"que tiene magia por"M(4):PRIN
T"fuerza de "F(4)"y sabiduria de "S(4):RETURN
350 PRINT"Esta habitacion contiene el mas":PRINT
"mortal enemigo de cuantos habitan el castillo":PR
INT"Se trata de Wolvling, el Hombre-Lobo, con":PRI
NT"una fuerza de "F(5):PRINT"una sabiduria de "S(5
):PRINT"y una capacidad magica de "M(5):RETURN
360 REM ****************************************
***************************
370 REM Final del juego
380 PRINT"Has venido a parar a los lodazales cie
nagosos que rodean el"
390 PRINT
400 PRINT"Lago Negro alrededor del castillo. Par
a escapar"
410 PRINT
420 PRINT"del castillo deberas luchar contra el
Guardian del Lago Negro"
430 PRINT
440 PRINT
450 PRINT"La lucha debe involucrar todos los atr
ibutos, y necesitaras"
460 PRINT
470 PRINT"      un total de 10 para escapar...."
480 GOSUB 4990:GOSUB 4990
490 IF DINERO>0 THEN PRINT "Tienes oro por valor
 de $"DINERO
500 GOSUB 4990
```

```
510 PRINT:PRINT"Los atributos del guardian:"
520 PRINT "      MAGIA:"M(6)
530 PRINT "     FUERZA:"F(6)
540 PRINT " SABIDURIA:"S(6)
550 PRINT:PRINT"Tus atributos son:"
560 PRINT "      MAGIA:"M(7)
570 PRINT "     FUERZA:"F(7)
580 PRINT " SABIDURIA:"S(7)
590 GOSUB 4990:GOSUB 4990
600 IF DINERO <100 THEN 790
610 PRINT:PRINT"Puedes mejorar tus atributos"
620 PRINT"Por $100 cada uno...."
630 PRINT"Si quieres comprar alguno, dime"
640 PRINT"la inicial del atributo
650 PRINT"seguida por el dinero que deseas"
660 PRINT"gastar para aumentar ese atributo."
670 PRINT"Dime 'N' cuando ya tengas"
680 PRINT"todos los atributos que deseas"
690 INPUT"Atributo (M, F o S)";E$
700 IF E$= "N" THEN 790
710 INPUT "CANTIDAD";CA
720 IF DINERO-CA<1 OR CA<100 THEN 710
730 DINERO=DINERO-CA
740 IF E$="M" THEN M(7)=M(7) +INT(CA/100)
750 IF E$="F" THEN F(7)=F(7)+INT(CA/100)
760 IF E$="S" THEN S(7)=S(7)+INT(CA/100)
                                    770 PRINT "Magia:"
M(7)" Fuerza:"F(7)
780 IF DINERO>99 THEN 690
790 CLS
800 PRINT"Ahora, la Ultima Prueba..."
810 GOSUB 4990
820 PRINT"Aprieta RETURN cuando estes "
830 INPUT"preparado para luchar",E$
840 GOSUB 4990:CLS
850 PRINT:PRINT"Primero, magia..."
860 PRINT:PRINT"Tu:"M(7)" Guardian:"M(6)
870 DIF=ABS(M(7)-M(6))
880 PRINT:PRINT"La diferencia es "DIF
890 IF M(7)>M(6) THEN PRINT" a tu favor"
900 IFM(6)>M(7) THEN PRINT"y el guardian te aven
taja."
910 GOSUB4990
```

```
  920 K=M(6)+M(7)
  930 PENA=INT(RND(-1)*K)+1
  940 PRINT"Esta lucha lleva una penalizacion"
  950 PRINT"de "PENA"puntos de atributo":GOSUB 499
0
  960 GOSUB 2600
  970 DIF=M(7)-M(6)
  980 IF DIF>5 THEN DIF=DIF-6:GOTO980
  990 IF DIF < -5 THEN DIF = DIF+6:GOTO990
 1000 GOSUB 190
 1010 IF VICTORIA =1 THEN M(7)=M(7)+PENA:PRINT"Y t
u has vencido...por lo tanto"
 1020 IF VICTORIA =0 THEN M(7)=M(7)-PENA:PRINT"Y h
as perdido...por lo tanto"
 1030 IFM(7)<1 THEN M(7)=0
 1040 PRINT "ahora tienes "M(7)"puntos de magia...
"
 1050 GOSUB 4990
 1060 PRINT"Oprime RETURN cuando estes dispuesto"
 1070 INPUT"para reanudar esta batalla epica";E$
 1080 CLS
 1090 PRINT"Ahora es el momento para una confronta
cion de "
 1100 PRINT"fuerza, donde tu puntuacion es "F(7)"y
 la del Guardian"
 1110 PRINT"es nada mas y nada menos que "F(6)
 1120 PRINT
 1130 DIF=ABS(F(7)-F(6))
 1140 PRINT"La diferencia es "DIF
 1150 IFF(6)>F(7) THEN PRINT"a favor del Guardian"
 1160 IFF(7)>F(6) THEN PRINT"a tu favor, "N$
 1170 GOSUB 4990
 1180 K=F(6)+F(7)
 1190 PENA=INT(RND(-1)*K)+1
 1200 PRINT:PRINT"Esta lucha tiene una penalizacio
n"
 1210 PRINT"de "PENA"puntos de atributo":GOSUB 499
0
 1220 DIF=F(7)-F(6)
 1230 GOSUB 2600
 1240 IF DIF>5 THEN DIF=DIF-6:GOTO1240
 1250 IF DIF<-5 THEN DIF=DIF+6:GOTO1250
 1260 GOSUB 190
```

```
1270 IF VICTORIA=1 THEN F(7)=F(7)+PENA:PRINT"Eres
 el vencedor, luego"
1280 IF VICTORIA=0 THEN F(7)=F(7)-PENA:PRINT"Has
perdido, luego"
1290 IFF(7)<0 THEN F(7)=0
1300 PRINT"ahora tienes "F(7)"puntos de fuerza"
1310 GOSUB 4990
1320 PRINT:PRINT"Oprime RETURN cuando"
1330 PRINT"hayas dejado de temblar"
1340 PRINT"lo suficiente para afrontar la tercera
, y"
1350 INPUT"ultima prueba..";E$
1360 GOSUB 4990
1370 CLS
1380 PRINT"Ahora es el momento para una prueba de
"
1390 PRINT"sabiduria, donde tu puntuacion "
1400 PRINT"es "S(7)"y la del Guardian"
1410 PRINT"es "S(6)
1420 DIF=ABS(S(7)-S(6))
1430 PRINT:PRINT"La diferencia es "DIF
1440 IFS(6)>S(7) THEN PRINT "a favor del guardian
"
1450 IFS(7)>S(6) THEN PRINT "a tu favor, "N$
1460 GOSUB 4990
1470 K=S(6)+S(7)
1480 PENA=INT(RND(-1)*K)+1
1490 PRINT"Ahora esta prueba final"
1500 PRINT"lleva una penalizacion extra "
1510 PRINT"de "PENA"puntos de atributo":GOSUB 499
0
1520 DIF=S(7)-S(6)
1530 IF DIF>5 THEN DIF=DIF-6:GOTO1530
1540 IF DIF<-5 THEN DIF=DIF+6:GOTO1540
1550 GOSUB 190
1560 GOSUB 2600
1570 IF VICTORIA=1 THEN S(7)=S(7)+PENA:PRINT"Y ve
ncistes al guardian!"
1580 IF VICTORIA=0 THEN S(7)=S(7)-PENA:PRINT"Pero
 el guardian lo hizo":PRINT"mejor que tu, "N$"!!"
1590 IFS(7)<0 THEN S(7)=0
1600 GOSUB 4990
1610 CLS:PRINT:PRINT
```

```
1620 PRINT"Y ahora, al final de la"
1630 PRINT"ultima batalla, tu posicion"
1640 PRINT"es:       Magia..."M(7)
1650 PRINT"     Sabiduria..."S(7)
1660 PRINT"         Fuerza..."F(7)
1670 SUM=M(7)+F(7)+S(7)
1680 PRINT:PRINT"Bien, "N$", tu"
1690 PRINT"total de atributos es "SUM
1700 GOSUB 4990
1710 IF SUM<10 THEN 1810
1720 PRINT"Necesitabas al menos 10 puntos"
1730 PRINT"para ganar el juego, y lo has"
1740 PRINT"conseguido, "N$"!":GOSUB 4990
1750 CLS:PRINT:PRINT:PRINT
1760 PRINT"Lo has conseguido, Oh heroe "
1770 PRINT"de estos oscuros y peligrosos"
1780 PRINT"tiempos.Yo, por lo tanto, os nombro"
1790 PRINT"Caballero "N$"........Por favor, levan
taos................."
1800 END
1810 PRINT"Desafortunadamente, no conseguisteis"
1820 PRINT"terminar con los 10 puntos"
1830 PRINT"que necesitabais, asi que todo ha term
inado."
1840 PRINT:PRINT"No lograstes escapar de las tram
pas"
1850 PRINT"del Guardian....":GOSUB 4990
1860 PRINT"Luchastes valientemente, pero ahora"
1870 PRINT"seras consumido....."
1880 GOSUB 4990:GOSUB 2600:GOSUB 2600:END
1890 REM ****************************************
**************************
1900 REM Accion
1910 D=4:IF MID$(B$(Z),9,1) ="O" OR MID$(B$(Z),9,
1) =" " THEN D=1:IF RND(-1)>.8 AND Z>1 THEN GOSUB
2740:GOTO1910
1920 PRINT"Que quieres hacer ahora";
1930 INPUTZ$:IFZ$="T"THEN END
1940 IFZ$="" THEN CLS:Z$="#"
1950 IFD=4 AND LEFT$(Z$,1)<>"L" THEN D=0:GOTO2090
1960 IF LEFT$(Z$,1)="L" THEN 2060
1970 IF Z$="N" AND LEFT$(B$(Z),2)="OO" THEN PRINT
,"No hay salida":GOTO1920
```

```
 1980 IFZ$="S" AND MID$(B$(Z),3,2)="00" THEN PRINT
"No hay puerta en esa direccion":GOTO1920
 1990 IFZ$="E" AND MID$(B$(Z),5,2) ="00" THEN PRIN
T "Eso no es posible":GOTO1920
 2000 IFZ$="O" AND MID$(B$(Z),7,2)="00" THEN PRINT
"No puedes pasar a traves de las paredes!":GOTO 19
20
 2010 IF Z$="N" THEN Z=VAL(LEFT$(B$(Z),2)):RETURN
 2020 IF Z$="S" THEN Z=VAL(MID$(B$(Z),3,2)):RETURN
 2030 IF Z$="E" THEN Z=VAL(MID$(B$(Z),5,2)):RETURN
 2040 IFZ$="O"  THEN Z=VAL(MID$(B$(Z),7,2)):RETURN
 2050 IF LEFT$(Z$,1)<>"L" THEN RETURN
 2060 IF RIGHT$(B$(Z),1)="0" THEN PRINT"No hay nad
a contra lo que luchar":GOTO1920
 2070 IF Z$="H" THEN D=INT(RND(-N)*2)
 2080 IF D=1 THEN PRINT"En que direccion?":GOTO193
0
 2090 IF D=0 THEN PRINT"NO!! Debes quedarte y luch
ar!"
 2100 PRINT:PRINT"Con que caracteristica lucharas?
"
 2110 PRINT"Tu magia es"M(7)", fuerza "F(7)", y tu
 sabiduria es"S(7)
 2120 INPUTZ$:IFZ$<>"M" AND Z$<>"F" AND Z$<>"S" TH
EN 2120
 2130 IF Z$="M" THEN HUM=M(7):BES=M(Q)
 2140 IF Z$="F" THEN HUM=F(7):BES=F(Q)
 2150 IF Z$="S" THEN HUM=S(7):BES=S(Q)
 2160 DIF=HUM-BES
 2170 IF DIF>5 THEN DIF=DIF-6:GOTO 2170
 2180 IF DIF <-5 THEN DIF= DIF+6:GOTO2180
 2190 PRINT"La tabla de lucha para este encuentro
indica "DIF
 2200 PENA = ABS(DIF) + INT(RND(-1)*6) + 1
 2210 GOSUB 4990
 2220 PRINT"Este encuentro conlleva un costo/recom
pensa de "PENA
 2230 FI= INT(RND(-1)*2):GOSUB 4990
 2240 IF FI=0 THEN PRINT "El monstruo ataca y la l
ucha esta teniendo lugar"
 2250 IF FI=1 THEN PRINT "Tu atacas primero y":PRI
NT"os enzarzais en la batalla"
 2260 GOSUB 2600
```

```
2270 PELEA=INT(RND(-1)*6)+1
2280 VICTORIA=0
2290 IF(DIF<0 AND PELEA>ABS(DIF)) OR (DIF>0  AND
PELEA<= DIF) OR (DIF=0 AND PELEA<4) THEN VICTORIA
=1
2300 IF VICTORIA=1 THEN GOSUB 2460
2310 IF VICTORIA=0 THEN GOSUB 2530
2320 GOSUB 4990
2330 PRINT"Despues de la lucha, tus"
2340 PRINT"atributos son:"
2350 PRINT"    Magia:"M(7)
2360 PRINT"   Fuerza:"F(7)
2370 PRINT"Sabiduria:"S(7)
2380 GOSUB 4990:PRINT
2390 PRINT" Y los de "M$(Q)" son:"
2400 PRINT"    Magia:"M(Q)
2410 PRINT"   Fuerza:"F(Q)
2420 PRINT"sabiduria:"S(Q)
2430 MID$(B$(Z),9,1) = "0"
2440 GOSUB 4990: GOSUB 4990:CLS
2450 GOTO2740
2460 REM Victoria humana
2470 IFQ=0 THEN D=1
2480 PRINT:PRINT"Vencistes al "M$(Q)" !!"
2490 IF Z$="M" THEN M(7) = M(7) +PENA:M(Q)=M(Q)-P
ENA:IFM(Q)<1 THEN M(Q)=0
2500 IF Z$="S" THEN S(7) = S(7) +PENA:S(Q)=S(Q)-P
ENA:IFS(Q)<1 THEN S(Q)=0
2510 IF Z$="F" THEN F(7) = F(7) +PENA:F(Q)=F(Q)-P
ENA:IFF(Q)<1 THEN F(Q)=0
2520 RETURN
2530 REM Victoria del monstruo
2540 PRINT:PRINT" El "M$(Q)" te ha vencido"
2550 IF Z$="M" THEN M(Q)=M(Q)+PENA:M(7)=M(7)-PENA
:IF M(7)<1 THEN M(7)=0
2560 IF Z$="S" THEN S(Q)=S(Q)+PENA:S(7)=S(7)-PENA
:IF S(7)<1 THEN S(7)=0
2570 IF Z$="F" THEN F(Q)=F(Q)+PENA:F(7)=F(7)-PENA
:IF F(7)<1 THEN F(7)=0
2580 RETURN
2590 REM ****************************************
***************************
2600 REM Efectos de lucha
```

```
2610 FOR J=1 TO RND(-1)*10+2
2620 ON (INT(RND(-1)*6 +1)) GOSUB 2670, 2680, 269
0, 2700, 2710, 2720
2630 FOR P=1 TO 100+RND(-N)*100:NEXT P
2640 PRINT:PRINT
2650 NEXT J
2660 CLS:RETURN
2670 PRINT"    Bash!!!!":FORP=1 TO 100:NEXT P:RET
URN
2680 PRINT,"Aaaaaarghhh!":FOR P=1 TO 20:NEXT P:RE
TURN
2690 PRINT"Rip":FOR P=1 TO 100:NEXTP:PRINT,,"CIEL
OS!":FOR P =1 TO 100 :NEXT P:RETURN
2700 FOR E=1 TO 20:PRINT"!!! ";:NEXT E:CLS:RETURN
2710 RETURN
2720 FOR E=1 TO 3:PRINT"!*&&*§!!      ";:NEXT E :F
OR P=1 TO 20:NEXT P:RETURN
2730 REM ****************************************
****************************
2740 REM Contenidos
2750 K=2+INT(RND(-1)*8)
2760 IF K=Z THEN 2750
2770 IF RIGHT$(B$(Z),1)<>"0" THEN RETURN
2780 MID$(B$(K),9,1)=MID$(B$(Z),9,1)
2790 MID$(B$(Z),9,1)="0"
2800 IF RND(-1)>.5 THEN RETURN
2810 PRINT
2820 CT=INT(RND(-1)*5)+1
2830 ON CT GOSUB 2870,2980,3100,3210,2870
2840 GOSUB 4990
2850 RETURN
2860 REM ****************************************
****************************
2870 COFRE=COFRE+1:IF COFRE=5 THEN RETURN
2880 PRINT"En frente tuyo hay un cofre"
2890 PRINT"etiquetado con un gran #"COFRE
2900 PRINT:PRINT"Vas a abrirlo?"
2910 GOSUB 3380
2920 IFZ$="N" THEN RETURN
2930 J=INT(RND(-1)*3):GOSUB 4990
2940 IF J =0 THEN BOLSA=100+INT(RND(-1)*300):PRIN
T"Contiene una bolsa con oro del dragon valorado e
n $"BOLSA:DINERO=DINERO+BOLSA:GOSUB4990:RETURN
```

```
 2950 IF J=1 THEN PRINT"Un duende salta del interi
or, y te hiere!":PERDIDA=INT(RND(-1)*6)+1:DINERO=D
INERO-INT(RND(-1)*200):IF DINERO<1 THEN DINERO=0:G
OSUB 4990:RETURN
 2960 IF J=2 THEN PRINT"Un estrano humo surge ":PR
INT"haciendo que te duermas y":PRINT"apoderandose
de tus poderes magicos":PERDIDA=INT(RND(-1)*8)+1:M
(7)=M(7)-PERDIDA:IF M(7)<1 THEN M(7)=0:GOSUB 4990:
RETURN
 2970 REM ****************************************
****************************
 2980 IF POCION=1 THEN 2820
 2990 POCION=1
 3000 PRINT"Ves una pequena botella, grabada"
 3010 PRINT"con unas curiosas letras retorcidas...
"
 3020 PRINT:PRINT"Quieres beber la pocion que "
 3030 PRINT"se puede ver en el interior de la bote
lla?"
 3040 GOSUB 3380
 3050 IF Z$="N" THEN RETURN
 3060 GOSUB 4990
 3070 IF RND(-1)>.6 THEN PRINT "contenia una pocio
n para":PRINT"aumentar tu sabiduria":S(7)=S(7)+INT
(RND(-1)*6)+1:GOSUB 4990:RETURN
 3080 PRINT"contenia una pocion que":PRINT"te debi
lita mas.....":GOSUB 4990:F(7)=F(7)-INT(RND(-1)*6
)-1:IFF(7)<1 THEN F(7)=0:GOSUB 4990:RETURN
 3090 REM ****************************************
****************************
 3100 IF CALAVERA=1 THEN RETURN
 3110 CALAVERA=1
 3120 PRINT"Ves un papiro enrollado."
 3130 PRINT"Quieres intentar leerlo?"
 3140 GOSUB 3380:IF Z$="N" THEN RETURN
 3150 IF RND(-1)>.5 THEN PRINT"No puedes entender"
:PRINT"el idioma...":GOSUB 4990:RETURN
 3160 PRINT"Contiene un conjuro magico.Quieres":PR
INT"leerlo?"
 3170 GOSUB 3380:IF Z$="N" THEN RETURN
 3180 IF RND(-1) > .5 THEN PRINT"Era un conjuro be
neficioso":GOSUB 4990:M(7)=M(7)+INT(RND(-1)*6)+1:R
ETURN
```

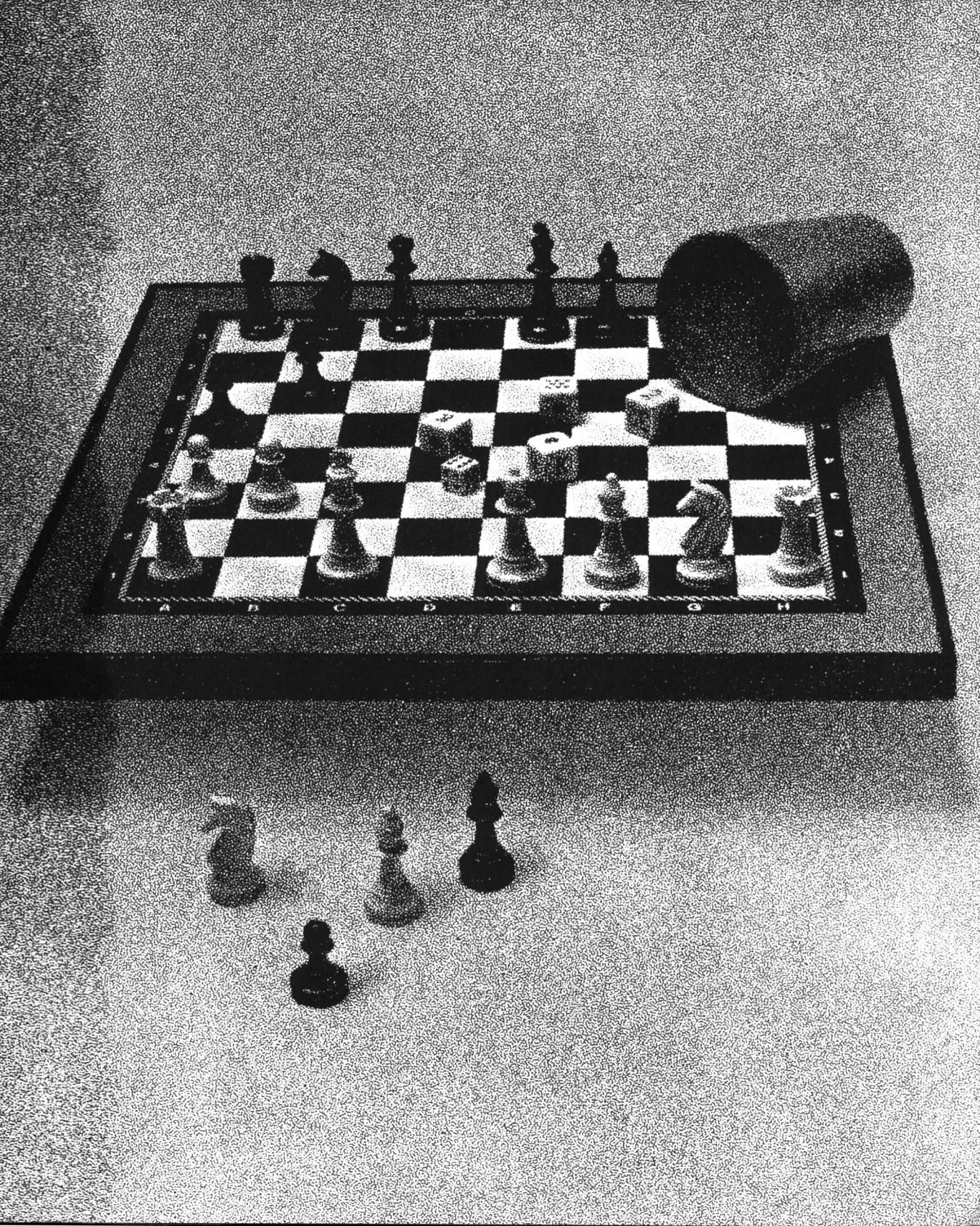
```
 3190 PRINT"Era un conjuro maligno":GOSUB 4990:M(7
)=0:F(7)=INT(F(7)/2):RETURN
 3200 REM ****************************************
****************************
 3210 IF CAJAFUERTE=1 THEN 2820
 3220 CAJAFUERTE=1
 3230 PRINT"En la pared hay una pequena caja fuert
e dorada":PRINT"y a su lado hay una llave..."
 3240 PRINT"Quieres abrir la caja fuerte?"
 3250 GOSUB 3380
 3260 IF Z$="N" THEN RETURN
 3270 IF RND(-1)>.3 THEN 3330
 3280 GOSUB 4990:PRINT"Una arpia ululante vuela ha
cia ti":PRINT"y unde sus dientes en ":PRINT"tu gar
ganta!!!"
 3290 GOSUB 4990
 3300 PRINT"Luchas con ella, y ...":GOSUB 4990:PRI
NT"... finalmente le retuerces el cuello"
 3310 F(7)=F(7)-INT(RND(-1)*6)-1:IFF(7)<1 THEN F(7
)=0
 3320 GOSUB 4990:RETURN
 3330 PRINT"Oyes un coro de":PRINT"voces angelical
es.....":GOSUB 4990
 3340 PRINT"Eres curado y refrescado..."
 3350 M(7)=M(7)+2:F(7)=F(7)+2:S(7)=S(7)+2
 3360 GOSUB 4990
 3370 RETURN
 3380 Z$=INKEY$
 3390 IFZ$<>"N" AND Z$<>"S" THEN 3380
 3400 PRINT:RETURN
 3410 REM ****************************************
*************************
 3420 REM DESCRIPCION DE LAS HABITACIONES
 3430 ON Z GOSUB 3480,3570,3630,3680,3770,3850,395
0,4080,4210,4270,4350,4440
 3440 Q=0:IF RND(-1)>.81 AND Z>1 THEN Q=INT(RND(-1
)*6):MID$(B$(Z),9,1)=MID$(STR$(Q),2,1):GOTO260
 3450 E$=MID$(B$(Z),9,1):IF E$>"0" AND E$<"6" THEN
 Q=VAL(E$)
 3460 IFQ>0 THEN GOSUB 260
 3470 RETURN
 3480 PRINT"Estas a la entrada de un"
 3490 PRINT"antiguo castillo de"
```

```
3500 PRINT"impresionante aspecto.Estas en"
3510 PRINT"el lado norte del castillo"
3520 PRINT"y al mirar hacia el sur, hacia"
3530 PRINT"la solida estructura, te"
3540 PRINT"das cuenta de que el portal de entrada
"
3550 PRINT"esta abierto y sin guardia"
3560 RETURN
3570 PRINT"Estas en el Hall de entrada,"
3580 PRINT"del que cuelgan ricos"
3590 PRINT"tapices.Las puertas dan hacia"
3600 PRINT"el este y el sur, y hay un"
3610 PRINT"portal abierto hacia el oeste"
3620 RETURN
3630 PRINT"Este es solo un pequeno almacen"
3640 PRINT"donde solo hay una salida, la "
3650 PRINT"misma por la que entrastes,"
3660 PRINT"hacia el oeste"
3670 RETURN
3680 PRINT"Esta pequena habitacion, que"
3690 PRINT"tiene una bonita escultura de"
3700 PRINT"la diosa de la luna en un "
3710 PRINT"pedestal en el rincon noreste,"
3720 PRINT"es la Real"
3730 PRINT"Camara de Recepciones.Hay puertas"
3740 PRINT"hacia el sur, el este, y "
3750 PRINT"el oeste....."
3760 RETURN
3770 PRINT"El Salon de las Intrigas, una"
3780 PRINT"habitacion con paneles de madera"
3790 PRINT"reminiscente de susurros y"
3800 PRINT"rumores, con salidas hacia el este"
3810 PRINT"y hacia el sur, de donde"
3820 PRINT"llega un olor de azufre y"
3830 PRINT"cantico magico...."
3840 RETURN
3850 PRINT"Has entrado a la cueva de "
3860 PRINT"las Brujas, con una marmita hirviendo"
3870 PRINT"en un fuego de llamas verdes"
3880 PRINT"en el rincon suroeste."
3890 PRINT"Esta habitacion rezuma azufre"
3900 PRINT"ardiente, y el eco de"
3910 PRINT"antiguos conjuros. Puedes salir"
```

```
3920 PRINT"hacia el norte, el sur, "
3930 PRINT"o el este"
3940 RETURN
3950 PRINT"Te encuentras en un lugar"
3960 PRINT"que parece tranquilo y pacifico."
3970 PRINT"Estas en la Galeria de Retratos"
3980 PRINT"del castillo,con un gran cuadro"
3990 PRINT"del Legendario Guardian"
4000 PRINT"del Lago Negro a la izquierda de"
4010 PRINT"la ventana de la pared este. A traves"
4020 PRINT"de la ventana puedes ver los parteluce
s"
4030 PRINT"de las ventanas del Gran Salon a trave
s de"
4040 PRINT"Jardin de los Contornos. Las salidas d
e la"
4050 PRINT"Galeria son hacia el norte y"
4060 PRINT"hacia el este....."
4070 RETURN
4080 PRINT"Este es el Salon mas suntuoso"
4090 PRINT"del castillo, el Gran Salon,"
4100 PRINT"con un impresionante techo"
4110 PRINT"labrado.Puedes salir por las puertas"
4120 PRINT"dobles del norte o por"
4130 PRINT"las del este, tras las cuales"
4140 PRINT"llega un sonido de musica."
4150 PRINT"A traves de las ventanas del muro"
4160 PRINT"oeste, puedes ver el Jardin"
4170 PRINT"de los Contornos, y mas alla,"
4180 PRINT"las ventanas de una habitacion con"
4190 PRINT"muchas, muchas pinturas valiosas."
4200 RETURN
4210 PRINT"Sonidos de un cuarteto de cuerda"
4220 PRINT"llenan esta habitacion, la Camara"
4230 PRINT"de los Musicos. Puedes salir por"
4240 PRINT"las puertas del oeste, o por"
4250 PRINT"la del sur..."
4260 RETURN
4270 PRINT"Ahora estas en el Santuario"
4280 PRINT"del Silencio, una habitacion cuya"
4290 PRINT"calma puede ser una decepcion."
4300 PRINT"La habitacion es humeda y fria. Una"
4310 PRINT"salida deja la habitacion hacia el"
```

```
4320 PRINT"norte ";
4330 IF MID$(B$(10),3,2)="12" THEN PRINT"y otra l
leva hacia el sur."
4340 PRINT:RETURN
4350 PRINT"Este debe ser el Vestibulo"
4360 PRINT"de las Visiones, una habitacion"
4370 PRINT"humeda y viscosa, donde la leyenda"
4380 PRINT"asegura que el Guardian del Lago Negro
"
4390 PRINT"a veces puede ser oido de noche."
4400 PRINT"Hay una puerta hacia"
4410 PRINT"el norte ";
4420 IF MID$(B$(11),3,2)="12" THEN PRINT "y otra
hacia el sur."
4430 PRINT:RETURN
4440 GOTO370
4450 IF MID$(B$(Z),9,1)<>"0" THEN Q=VAL(MID$(B$(Z
),9,1):GOSUB 260
4460 RETURN
4470 REM ****************************************
****************************
4480 REM INICIALIZACION
4490 RANDOMIZE
4500 Z=1
4510 DINERO=0
4520 COFRE=0
4530 POCION=0
4540 CALAVERA=0
4550 CAJAFUERTE=0
4560 DIM B$(12),M$(6),M(7)
4570 CLS:PRINT:PRINT
4580 INPUT"POR FAVOR, TECLEA TU NOMBRE";N$
4590 CLS:PRINT"HOLA, ";N$
4600 PRINT"UN MOMENTO, POR FAVOR..."
4610 REM Llena las habitaciones
4620 FOR T=1 TO 12
4630 READB$(T)
4640 L=RND(-1)
4650 Q$=STR$(INT(RND(-1)*5)+1)
4660 IF T>1 AND T<11 AND L<.63 THEN B$(T) =B$(T)
+ RIGHT$(Q$,1)
4670 IF T>10 OR L>=.63 THEN B$(T)=B$(T)+"0"
4680 NEXT T
```

```
4690 L=RND(-1)
4700 IF L<.5 THEN B$(11)= "091200000"
4710 IF L>=.5 THEN B$(10)="061200000"
4720 REM Crea los monstruos
4730 FOR T=1 TO 6
4740 READ M$(T)
4750 F(T)=INT(RND(-1)*6)+1
4760 M(T)=INT(RND(-1)*6)+1
4770 S(T)=INT(RND(-1)*6)+1
4780 NEXT T
4790 REM Caracteristicas humanas
4800 F(7)=INT(RND(-1)*6)+1
4810 M(7)=INT(RND(-1)*6)+1
4820 S(7)=INT(RND(-1)*6)+1
4830 REM Datos de las habitaciones
4840 DATA"000200000"
4850 DATA"00080304"
4860 DATA"00000002"
4870 DATA"00070205"
4880 DATA"00060400"
4890 DATA"05100700"
4900 DATA"04000600"
4910 DATA"02000900"
4920 DATA"00110008"
4930 DATA"06000000"
4940 DATA"09000000"
4950 DATA"00000000"
4960 REM Nombres de las bestias
4970 DATA"Mastin","Hidra","Demonio","Quebrantahue
sos","Wolvling","Guardian"
4980 REM ****************************************
*************************
4990 REM Rutina de pausa
5000 FOR P=1 TO  500:NEXT P
5010 PRINT
5020 RETURN
```


A B C D E F G H

SIMULACIONES

Las simulaciones son modos de producir una realidad falsa. En comparación con las AVENTURAS, en las cuales la realidad que se crea por medio del ordenador suele ser mágica y parecida a un sueño, los mundos a los que uno puede acceder por medio de simulaciones en el ordenador son, en general, más terrenales.

En una simulación, el ordenador manipula variables de acuerdo con fórmulas que uno le ha especificado previamente, no perdiendo de vista ni un momento el desarrollo de la situación que el programador ha creado y ocupando el lugar del entorno en términos de reaccionar a la información que se le introduce.

Las simulaciones intentan hacer una réplica de la vida real. Sin embargo, como la realidad es muy difícil de limitar y restringir, hay que hacer una simplificación bastante drástica antes de poder producir una imitación que pueda funcionar.

A pesar de esta simplificación, como verá en seguida, una simulación bien escrita puede remedar la vida real hasta unos extremos pavorosos. Causa y efecto están ligados de un modo tan preciso como un comportamiento que siguiera fórmulas ideadas por el jugador. El generador de números aleatorios puede ocupar el lugar de cosas tales como los cambios de tiempo, el número de habitantes, el comportamiento de las moléculas en el gas o el modo en que una tinta de color se dispersa en un líquido transparente.

En este capítulo del libro se incluyen tres simulaciones. Una de ellas —LA SEÑORA XENOFOBIA— es tan simple que llega al absurdo, pero es un juego la mar de divertido, ya que el jugador intenta remendar el destino de todo un planeta. Los otros dos

—PRACTICAS RURALES (en el que el jugador lleva una granja cuyos jornaleros son de lo más veleidoso) y PRESIDENTE DEL CONSEJO DE ADMINISTRACION (en el que el jugador administra una fábrica que podría producir cualquiera de una gran variedad de productos desde bicicletas a libros, pasando por cascos de bomberos)— son un poco más serios y las fórmulas empleadas en ellos producen resultados un poco más próximos a la «vida real».

A pesar de su «seriedad» estos dos últimos son también muy divertidos, como comprobará en cuanto los haya ejecutado una vez.

Si quiere más información sobre el modo de escribir programas de simulaciones, el folleto *Designing Classroom Simulations*, escrito por Glenn Pate y Hugh Parker, Jr., le será de una gran ayuda. Asimismo la revista *Creative Computing* ha publicado dos valiosos artículos sobre este tema. Estos son «How to write a computer simulation», cuyo autor es el propio editor de la revista, David Ahl, aparecido en el número de enero/febrero de 1978 (y que es el texto de una conferencia que Ahl dio en junio del 74 en un congreso sobre ordenadores que tuvo lugar en Pullman, Washington) y «Strategies for Successful Simulation», de Bruno B. Wolff, Jr., que apareció en el número de agosto de 1981. Merece la pena que consiga estos números atrasados ya que aprenderá mucho con la lectura de ambos artículos. De particular interés en el artículo de Wolff es el conjunto de cuatro ecuaciones que ligan causa y efecto y que pueden adaptarse muy fácilmente a cualquier programa de imitación que uno quiera escribir.

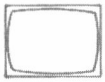
LA SEÑORA XENOFOBIA

Empezaremos nuestra exploración de las imitaciones con el programa más disparatado de los tres, LA SEÑORA XENOFOBIA, que está basado en un programa escrito por Alastair Gourlay, un programador de gran talento que vive en Glasgow.

No hay ninguna necesidad de explicar este juego, ya que el programa le da todo lo que el jugador puede necesitar. Aquí se le ofrece una parte de la ejecución del programa:

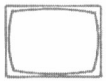
```
Senor de Xenophobia, un dossier para
Su Xenelencia de la Oficina de Informacion
referente al estado de su planeta
en el ano de Gracia de  1995


La poblacion del planeta es  432

Los viles campesinos podrian trabajar
unos  116  acres este ano ...

El tesoro contiene oro, gemas
y monedas por valor de $ 945

Es hora ya de promulgar un decreto...

Cuanta tierra va a ser cultivada
este ano ?27
```

Los viles campesinos podrian trabajar
unos 99 acres este ano ...

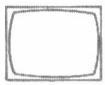


El tesoro contiene oro, gemas
y monedas por valor de $ 1021

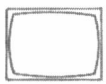
Es hora ya de promulgar un decreto...



Cuanta tierra va a ser cultivada
este ano ?24

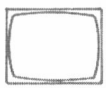


Y cuanto vas a gastar
en comida para los campesinos ?24

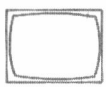


Oh, cielos!! No hay comida para todos...

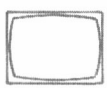
Que horror, su Xenolteza



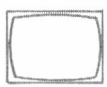
Ha habido una rebelion!!!
Tus campesinos estan descontentos
y se han sublevado.



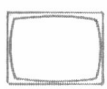
El tesoro contiene oro, gemas
y monedas por valor de $ 845



Es hora ya de promulgar un decreto...



Cuanta tierra va a ser cultivada
este ano ?100



Bien, Xeno, en bonito lio
te has metido tu y nuestro
pequeno planeta.

El Tesoro esta en bancarrota!!

Sabes quien fue el culpable??????????????

Si quieres ser reelegido,
limitate a apretar la 'S'.
O bien la 'N' si quieres terminar esta farsa...

Y este es el listado:

```
   10 REM La senora Xenofobia
   20 GOSUB 700:REM Inicializacion
   30 REM ****************************************
**************************
   40 FOR Y = 1 TO 20
   50 CLS
   60 PRINT:PRINT:PRINT
   70 PRINT"Senora de Xenofobia, un dossier para"
   80 PRINT"Su Xenelencia de la Oficina de Informa
cion"
   90 PRINT"referente al estado de su planeta"
  100 PRINT"en el ano de Gracia de "1994 + Y
  110 PRINT:PRINT
  120 PRINT"La poblacion del planeta es "INT(P)
  130 GOSUB 880
  140 PRINT"Los viles campesinos podrian trabajar"
  150 PRINT"unos "L" acres este ano ..."
  160 GOSUB 880
  170 PRINT"El tesoro contiene oro, gemas"
  180 PRINT"y monedas por valor de $"INT(U)
  190 GOSUB 880
  200 PRINT"Es hora ya de promulgar un decreto..."
  210 PRINT:PRINT"Cuanta tierra va a ser cultivada
"
  220 INPUT"este ano ";W
  230 U=U-W*10
  240 IF U < 1 THEN 780
  250 L=L+W
```

```
260 GOSUB 880
270 PRINT:PRINT"Y cuanto vas a gastar"
280 INPUT"en comida para los campesinos ";W
290 U=U-W*10
300 IF U<1 THEN 780
310 R=INT(RND(-1)*10)+1
320 GOSUB 880
330 IF P-W*R*5> P/4 THEN PRINT"Oh, cielos!! No h
ay comida para todos...":GOSUB 580
340 P=P+RND(-1)*(W*R*5- P)
350 IF P> 149 THEN 410
360 GOSUB 880
370 PRINT"La poblacion de Xenofobia es"
380 PRINT"ahora de solo "P" y eso"
390 PRINT"no es suficiente, Alteza."
400 GOTO480
410 U=U+INT(P*L)/93
420 NEXT Y
430 PRINT:PRINT:PRINT"Bien, Oh Gran Xenophobica,
 este es"
440 PRINT"el fin de tu dominio de 20 anos sobre"
450 PRINT"nuestro pequeno planeta. Conseguistes"
460 PRINT"acumular unos $"U", lo que"
470 PRINT"no esta nada mal..."
480 GOSUB 880
490 PRINT:PRINT"Si quieres ser reelegida,"
500 PRINT"limitate a apretar la 'S'."
510 PRINT"O bien la 'N' si quieres terminar esta
 farsa..."
520 A$=INKEY$
530 IF A$ <> "N" AND A$<>"n" AND A$<>"S" AND A$<
>"s" THEN 520
540 IF A$="N" OR A$="n" THEN PRINT"Tus deseos so
n ordenes para mi,":PRINT"Oh Senora de Xenofobia":
END
550 GOTO 10
560 REM ****************************************
*************************
570 REM Rebelion
580 PRINT:PRINT"Que horror, su Xenolteza"
590 PRINT"Ha habido una rebelion!!!"
600 PRINT"Tus campesinos estan descontentos"
610 PRINT"y se han sublevado."
```

```
620 GOSUB 880
630 IF RND(-1) <.2 THEN PRINT "Y tus guardas no
han podido controlarlos":GOTO 480
640 PRINT"Pero tus guardas han acabado"
650 PRINT"con toda esa falta de sentido."
660 GOSUB 880
670 RETURN
680 NEXT Y
690 REM ****************************************
*************************
700 REM Inicializacion
710 RANDOMIZE
720 CLS
730 P=200+INT(RND(-1)*300)
740 U=700+INT(RND(-1)*500)
750 L=70 +INT(RND(-1)*50)
760 RETURN
770 REM ****************************************
**************************
780 REM Bancarrota
790 PRINT:PRINT"Bien, Xeno, en bonito lio"
800 PRINT"te has metido tu y nuestro"
810 PRINT"pequeno planeta."
820 GOSUB 880
830 PRINT:PRINT"El Tesoro esta en bancarrota!!"
840 GOSUB 880
850 PRINT"Sabes quien fue el culpable??????????
?????"
860 GOTO490
870 REM ****************************************
**************************
880 REM Retardo
890 FOR J=1 TO 1500:NEXT J
900 PRINT
910 RETURN
```

PRACTICAS RURALES

Si ha sobrevivido a LA SEÑORA XENOFOBIA, puede que le interese ahora algo más pegado a la tierra: llevar una granja. En PRACTICAS RURALES, que está ligeramente basado en un programa escrito por Stephen Glen, también de Glasgow, el jugador se enfrenta a una difícil tarea, ya que ha de vérselas con un grupo de jornaleros extremadamente susceptibles para llevar a casa el tocino (o al menos el trigo y la cebada).

Nos encontramos de nuevo ante un programa que en gran medida funciona solo. Sencillamente trabajando con él aprenderá el lector cuál es la mejor estrategia para conseguir sacarle a la granja el máximo de ganancias (al tiempo que lograr sobrevivir los veinte años exigidos para ganar la partida). Sin embargo, no vendrán mal algunas sugerencias. Aunque es cierto que puede ahorrar dinero pagándoles a sus trabajadores un sueldo muy bajo, no se fíe de esto, ya que pueden irse en bloque al año siguiente y las ganancias que se sacan de la tierra vienen dictadas, hasta cierto punto, por el número de personas que se tienen trabajándola.

Otro factor que no se debe olvidar es que las tres cosechas que el jugador puede plantar producen ganancias bastante diferentes, así pues, le interesa calcular cuál es la cosecha que más beneficio produce para posteriormente concentrarse en ella.

No se desanime si las primeras veces que ejecuta el programa echa a perder el proceso. Tendrá que desarrollar técnicas reales para mantener la granja solvente y puede llevarle varios malos años hasta que le coja «el tranquillo».

Lo que viene a continuación es una parte de la salida de PRACTICAS RURALES:

```
Tienes  7231  pts. en el ano  1

Estas empleando  926
trabajadores para  212  acres

Cosechas:-      0  de maiz
                0 de cebada
                0  de trigo

Gastos generales por cada acre trabajado:  10  pts.
El maximo numero de acres que puedes trabajar es:  212
Cuanta tierra quieres arar?100
```

```
Tienes  6349  pts. en el ano  1

Estas empleando  675
trabajadores para  258  acres

Cosechas:-      0  de maiz
                0 de cebada
                0  de trigo

Cuanto pagaras a cada trabajador?11
```

```
Tienes  5482  pts. en el ano  1

Estas empleando  107
trabajadores para  424  acres

Cosechas:-      0  de maiz
                0  de cebada
                0  de trigo
Que proporcion (de 10)
Quieres concentrar en el maiz?3
```

```
Tienes  508  pts. en el ano  2

Estas empleando  420
trabajadores para  485  acres

Cosechas:-      100  de maiz
                12 de cebada
                158  de trigo

Se cosecharon  270  toneladas

Y tu ganancia total
fue de  2093  pts.
```

Y aquí tiene el listado que le convertirá en un granjero feliz:

```
 10 REM Practicas rurales
 20 RANDOMIZE
 30 YR=1
 40 MO=INT(RND(1)*1000+7000)
 50 LA=INT(RND(1)*1000+100)
 60 AC=INT(RND(1)*300+200)
 70 CS=INT(RND(1)*5)+8:CX=CS
 80 BA=0:WH=0:CO=0
 90 GOSUB 170
100 GOSUB 290
110 IF YR=10 THEN 750
120 IF MO<1 THEN 790
130 IF LA<1 THEN 820
140 YR=YR+1
150 CS=CS+INT(12.5*CS/100):REM Los costes genera
les se incrementan en un 12,5 % anual
160 GOTO 80
170 REM Actualizacion
180 GOSUB 870
```

```
190 PRINT "Tienes "MO" pts. en el ano "YR
200 GOSUB 880
210 PRINT "Estas empleando "LA
220 PRINT "trabajadores para "AC" acres"
240 GOSUB 880
250 PRINT "Cosechas:-", CO" de maiz"
260 PRINT ,BA "de cebada"
270 PRINT ,WH" de trigo"
280 RETURN
290 PRINT:PRINT "Gastos generales por cada acre
trabajado: "CS" pts."
300 PRINT "El maximo numero de acres que puedes
trabajar es: ";
340 MAX=INT(MO/CS):IF MAX > AC  THEN MAX=AC
350 PRINT MAX
360 PRINT"Cuanta tierra quieres arar";
370 INPUT L
380 IF L> MAX THEN 370
390 MO=MO-L*CS
400 GOSUB 170
410 PRINT:PRINT "Cuanto pagaras a cada trabajado
r";
420 INPUT W
430 IF W*LA>MO THEN 420
440 MO=MO-LA*W
450 GOSUB 170
460 P=10
470 PRINT "Que proporcion (de 10) "
480 PRINT "Quieres concentrar en el maiz";
490 INPUT CP
500 IF CP>P THEN 490
510 P=P-CP
520 PRINT:PRINT "De los "P" restantes de 10, "
530 PRINT "cuanto trigo vas a plantar";
540 INPUT WP
550 IF WP>P THEN 540
560 P=P-WP
570 GOSUB 870
580 PRINT "Espera para que transcurra un periodo
 anual..."
590 FOR Z=1 TO 2000: NEXT Z
600 BA=INT (P*L*LA*W*3/100000)
610 CO=INT (CP*L*LA*W*2.7/17000)
```

```
620 WH=INT (WP*L*LA*W*1.4/9300)
630 T=BA+CO+WH
640 GOSUB 170
650 PRINT:PRINT "Se cosecharon "T" toneladas"
660 RT=INT ((.5+8.7*BA+5.94*CO+2.2*WH)*(CS-CX+1)

670 IF BA=0 AND CO=0 AND WH=0 THEN RT=0
680 FOR Z=1 TO 2000: NEXT Z
690 PRINT:PRINT "Y tu ganancia total "
700 PRINT "fue de "RT" pts."
710 MO=MO+RT
720 LA=INT(LA-LA/(W+.01))
730 FOR Z=1 TO 2000:NEXT Z
740 RETURN
750 GOSUB 870
760 PRINT "Has sobrevivido durante 10 anos"
770 PRINT "Felicidades"
780 END
790 GOSUB 870
800 PRINT "Has llegado a la bancarrota!"
810 END
820 GOSUB 870
830 PRINT "No tienes trabajadores y"
840 PRINT "has tenido que vender"
850 PRINT ,"tu granja"
860 END
870 CLS
880 PRINT:PRINT
890 RETURN
```

PRESIDENTE DEL CONSEJO DE ADMINISTRACION

Por último tenemos la imitación más importante de las tres, PRESIDENTE DEL CONSEJO DE ADMINISTRACION.

Este programa también funciona solo en su mayoría. El objetivo del juego es lograr mantener en funcionamiento una fábrica hasta que se hayan ganado 10.000 dólares (en total, combinando el valor de los artículos que se tienen en «stock» y el capital). El jugador tendrá que entendérselas con los sindicatos, a veces recalcitrantes (que no le permitirán hacer despidos improcedentes y que estarán continuamente exigiendo subidas salariales a las que el jugador no se podrá negar), con unos obreros que a duras penas consiguen cumplir los índices de producción que se les marcan y con unos proveedores de materias primas que disfrutan tanto subiendo los precios como los sindicatos presionándole para que aumente los salarios.

El jugador verá en seguida, en contraste con los otros dos programas de este capítulo que no son sino vagas aproximaciones a la vida real, que la vida y el estado económico de la fábrica se convertirán en algo muy importante para él, y es seguro que no tardará en aprender la manera de manipular los recursos con los que cuenta para conseguir un máximo de beneficios. No debe olvidar el jugador que, aunque tiene toda la libertad para fijar el precio al detalle de los artículos, cada subida de precio aumentará la resistencia del consumidor a la hora de comprar esos productos.

Estos son unos cuantos ejemplos de una partida:

```
Informe de las tiendas, senor,
     para la semana No  1

El capital a mano es $ 831
Los costos de mantenimiento son $ 108 a la semana

Tus departamentos tienen  204 porrones
     valorados en $ 2856

Se estan vendiendo a $ 14  cada uno
y cuesta $ 10  fabricarlos.

Estas empleando ahora unos
 11  trabajadores,y les
pagas $ 20  a cada uno, asi que
los salarios te costaran esta semana $ 220

Cada persona puede hacer  10
porrones a la semana, lo que hace
un total de  110
Cuanta gente quieres
contratar?3

   Tus departamentos tienen  350 pucheros
        valorados en $ 3500

   Se estan vendiendo a $ 10  cada uno
   y cuesta $ 9  fabricarlos.

   Estas empleando ahora unos
    15  trabajadores,y les
   pagas $ 13  a cada uno, asi que
   los salarios te costaran esta semana $ 195
```

```
Cada persona puede hacer  9
pucheros a la semana, lo que hace
un total de  135
Cuantos quieres producir?100

No tienes suficiente dinero
Cuantos quieres producir?20

Si, senor... el objetivo  de esta semana No  1
es  20 pucheros
```

```
El capital a mano es $ 560
Los costos de mantenimiento son $ 110 a la semana

Tus departamentos tienen  559 pantalones de pana
     valorados en $ 7826

Se estan vendiendo a $ 14  cada uno
y cuesta $ 11  fabricarlos.

Estas empleando ahora unos
 12  trabajadores,y les
pagas $ 16  a cada uno, asi que
los salarios te costaran esta semana $ 192

Cada persona puede hacer  10
pantalones de pana a la semana, lo que hace
un total de  120

Tu stock total de
pantalones de pana es  559
```

Y ahora el lector ya puede satisfacer sus ambiciones mercantiles con este listado:

```
 10 REM Presidente del consejo de administracion
 20 GOSUB 1670:REM inicializacion
 30 SE = SE + 1
 40 GOSUB 930: REM report
 50 GOSUB 1300: REM plantilla
 60 GOSUB 930: REM informe
 70 GOSUB 1130: REM produccion
 80 GOSUB 930: REM informe
 90 GOSUB 730: REM ventas
100 GOSUB 140: REM problemas
110 CA= CA - SA*TR - CM
120 GOTO 30
130 REM ****************************************
************************
140 REM problemas
150 CLS
160 IF RND(-1) < .45 THEN 260
170 A= INT( RND(-1) * 7) + 1
180 PRINT: PRINT: PRINT
190 PRINT"Los sindicatos demandan un"
200 PRINT"aumento en el salario del "A"%"
210 SA = INT(100*(SA + SA*A/100))/100
220 GOSUB 1840
230 PRINT:PRINT"El salario por empleado es ahora
$"SA
240 GOSUB 1840
250 CLS
260 IF RND(-1) < .81 THEN 410
270 PRINT:PRINT:PRINT
280 PRINT"Un fuego en tus almacenes ha"
290 PRINT"destruido parte del stock.Por favor"
300 PRINT"aguarda un informe de los"
310 PRINT"danos causados..."
320 GOSUB 1840
330 A=INT(RND(-1) * DE/2) + 1
340 DE = DE - A
350 PRINT:PRINT"Ha habido"A;A$
360 PRINT"destruidos.Estaban"
```

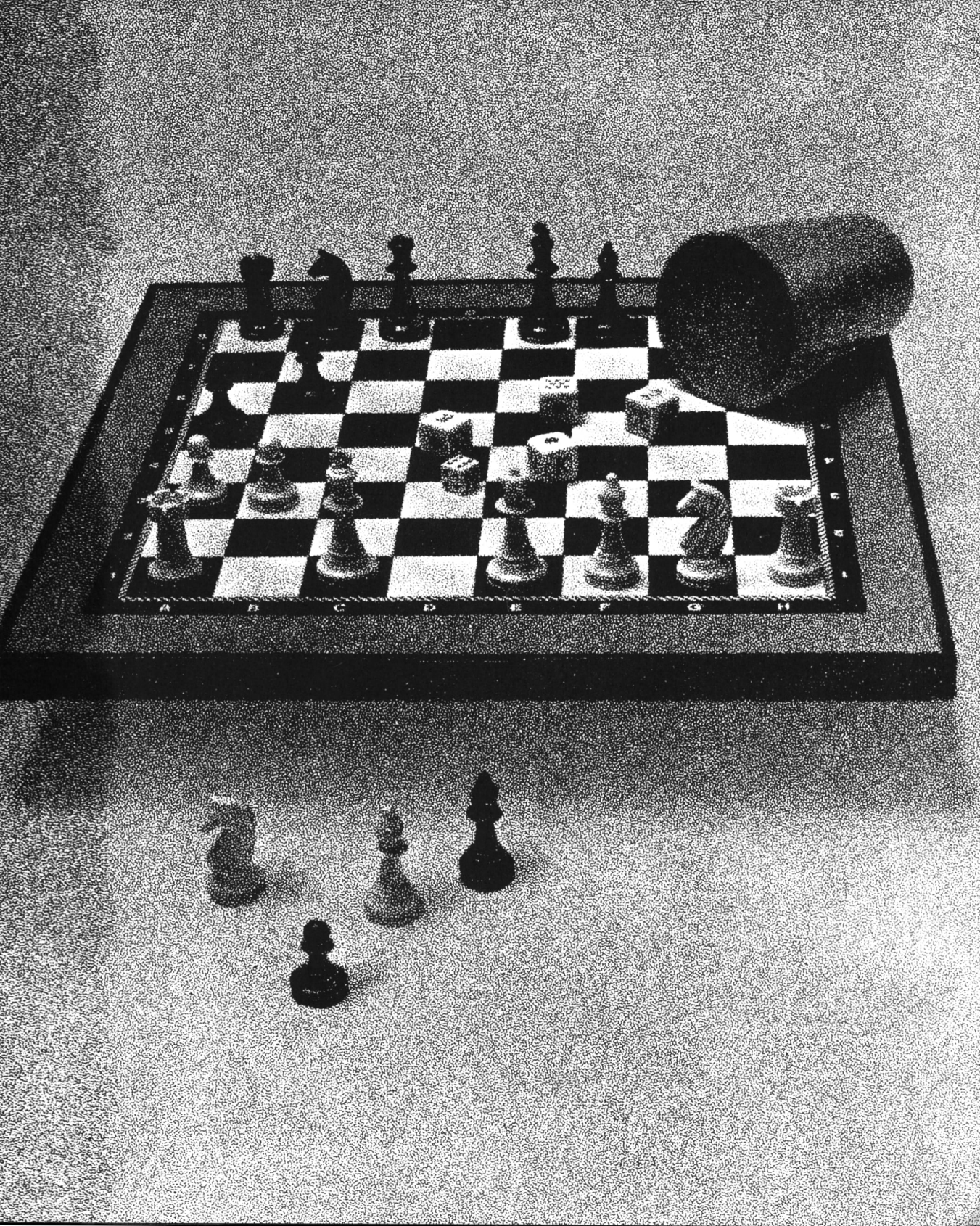
```
370 PRINT"valorados en $"A*PV" P.V.P."
380 GOSUB 1840
390 PRINT"El stock que pooseemos"
400 PRINT"ahora es de "ST;A$
410 IF RND(-1) >.3 THEN 560
420 CLS
430 PRINT:PRINT:PRINT
440 PRINT"Tu principal proveedor ha anunciado"
450 PRINT"un dramatico incremento..."
460 GOSUB 1840
470 A=INT(RND(-1)*100*CO/7)/100
480 IF A<.01 THEN 470
490 PRINT:PRINT"El costo de fabricacion de "A$
500 PRINT"se ha incrementado en $"A"por unidad"
510 GOSUB 1840
520 CO = CO + A
530 PRINT:PRINT"Ahora cuestan $"CO
540 PRINT"por unidad..."
550 GOSUB 1840
560 IF RND(-1)< 0.65 AND CU < PV THEN RETURN
570 CLS
580 PRINT:PRINT:PRINT
590 PRINT"Tienes una oportunidad de subir"
600 PRINT TAB(4);"el precio. Ahora"
610 PRINT"estas vendiendo tus "A$" a $"PV
620 GOSUB 1840
630 PRINT
640 PRINT"Que porcentaje de incremento"
650 INPUT"te gustaria imponer";A
660 RE = RE + A
670 PV = INT(100*(PV + A*PV/100))/100
680 GOSUB 1840
690 PRINT:PRINT"Tus "A$" se estan vendiendo ahor
a por $"PV
700 GOSUB 1840
710 RETURN
720 REM ****************************************
****************************
730 REM ventas
740 PRINT:PRINT"Tu stock total de"
750 PRINTA$" es "DE
760 GOSUB 1840
770 PRINT:PRINT"Por favor, aguarda un"
```

```
  780 PRINT"informe de ventas..."
  790 A=INT(RND(-1)*DE/(RE/1000)) + 1
  800 IF A > DE THEN 790
  810 CLS
  820 PRINT:PRINT:PRINT
  830 PRINT"El numero total de "A$
  840 PRINT"vendido es "A
  850 DE = DE - A
  860 ZA = A*PV
  870 PRINT:PRINT"El producto de"
  880 PRINT"esa venta fue $"ZA
  890 CA = INT(A*PV*100)/100 + CA
  900 GOSUB 1840
  910 RETURN
  920 REM ****************************************
***************************
  930 REM Informe al Gerente
  940 CLS
  950 IF CA + DE < 1 THEN 1510:REM bancarrota
  960 IF CA + DE > 9999 THEN PRINT "Has conseguido
 mas de $10.000 y":PRINT"puedes ya retirarte...":G
OSUB 1840: GOTO 1590
  970 PRINT:PRINT"Informe de las tiendas, senor,"
  980 PRINT TAB (6);"para la semana No "SE
  990 PRINT:PRINT"El capital a mano es $"INT(CA*10
0)/100
 1000 PRINT"Los costos de mantenimiento son $"CM"a
 la semana"
 1010 PRINT:PRINT"Tus departamentos tienen "DE;A$
 1020 PRINT TAB(6);"valorados en $"INT(DE*PV*100)/
100
 1030 PRINT:PRINT"Se estan vendiendo a $"PV" cada
uno"
 1040 PRINT"y cuesta $"CO" fabricarlos."
 1050 PRINT:PRINT"Estas empleando ahora unos"
 1060 PRINT TR" trabajadores,y les"
 1070 PRINT"pagas $"SA" a cada uno, asi que"
 1080 PRINT"los salarios te costaran esta semana $
"SA*TR
 1090 PRINT:PRINT "Cada persona puede hacer "PR
 1100 PRINT A$" a la semana, lo que hace"
 1110 PRINT"un total de "PR*TR
 1120 RETURN
```

```
1130 INPUT "Cuantos quieres producir";CU
1140 IF CU = 0 THEN RETURN
1150 PRINT
1160 IF CU*CO > CA THEN PRINT "No tienes suficien
te dinero":GOTO1130
1170 IF CU> PR*TR THEN PRINT"No tienes suficiente
 gente":PRINT"en tu plantilla para hacer":PRINT TA
B(6); "todos esos": GOTO 1130
1180 PRINT"Si, senor... el objetivo  de esta sema
na No "SE
1190 PRINT"es "CU;A$
1200 CU = INT(CU - RND(-1)*CU/5)
1210 GOSUB 1840
1220 PRINT:PRINT"El numero de "A$
1230 PRINT"que hemos conseguido fabricar en esta
semana No "SE
1240 PRINT"fue de "CU"..."
1250 DE = DE + MA
1260 CA = CA -CO*CU
1270 GOSUB 1840
1280 RETURN
1290 REM ****************************************
*************************
1300 REM plantilla
1310 PRINT"Cuanta gente quieres"
1320 INPUT"contratar";A
1330 TR = TR + A
1340 PRINT:PRINT"La plantilla total es ahora"
1350 PRINT"de unos "TR" empleados"
1360 GOSUB 1840
1370 IF A > 0 THEN RETURN
1380 GOSUB 930
1390 PRINT"A cuanta gente"
1400 INPUT"quieres despedir";A
1410 IF A = 0 THEN 1480
1420 IF A > TR THEN 1390
1430 A=INT(RND(-1)*A + 1)
1440 GOSUB 1840
1450 PRINT:PRINT "Los sindicatos te permitiran"
1460 PRINT"despedir a "A
1470 TR = TR - A
1480 GOSUB 1840
1490 RETURN
```

```
 1500 REM ****************************************
**************************
 1510 REM bancarrota
 1520 PRINT:PRINT"Estas en bancarrota!!!"
 1530 GOSUB 1840
 1540 PRINT:PRINT"Oh, que pena!!!"
 1550 GOSUB 1840
 1560 PRINT:PRINT"Aun asi, coseguistes que el nego
cio"
 1570 PRINT"marchara durante "SE" semanas"
 1580 GOSUB 1840
 1590 PRINT"Aprieta la 'S' si quieres volver a"
 1600 PRINT"hacer de Gerente ..."
 1610 PRINT"(o 'N' si quieres dejarlo)"
 1620 A$=INKEY$
 1630 IFA$="" THEN 1620
 1640 IF A$="S" OR A$="s" THEN GOTO 10
 1650 END
 1660 REM ****************************************
***************************
 1670 REM inicializacion
 1680 RANDOMIZE
 1690 FOR Z= 1 TO RND(-1)*8 + 1
 1700 READ A$
 1710 NEXT Z
 1720 CA = 500 + INT(RND(-1)*500)
 1730 DE = 100 + INT(RND(-1)*500)
 1740 PV = 10 + INT(RND(-1)*5)
 1750 CO = 7 + INT(RND(-1)*5)
 1760 IF CO > PV THEN 1750
 1770 TR = 7 + INT(RND(-1)*10)
 1780 SA = 12 + INT(RND(-1)*PV)
 1790 PR = 5 + INT(RND(-1)*6)
 1800 CM = 100 + INT(RND(-1)*20)
 1810 SE = 0
 1820 RE = 1:REM RE es el factor de resistencia a
las ventas
 1830 RETURN
 1840 REM subrutina de retardo
 1850 FOR Z = 1 TO 3000:NEXT Z:RETURN
 1860 DATA"libros de cuentos","motocicletas","trom
bones"
 1870 DATA"pucheros","extintores","porrones"
```

```
1880 DATA"pantalones de pana","cascos de bombero"
,"anzuelos"
1890 DATA"diccionarios"
```


A B C D E F G H

JUEGOS DE DADOS

Los dados han sido y siguen siendo uno de los juegos más populares que hayan existido –ya se apueste alto, bajo o nada—. Independientemente de que estén hechos de madera o piedra o labrados en los huesos de los animales matados para alimento, de que tengan cuatro o seis caras, los dados han sido un motivo de diversión para una incontable cantidad de personas a lo largo de la historia.

Los dados, un medio físico de demostrar la indocilidad de la suerte, se emplearon y se emplean para predecir el futuro y, como todos los oráculos, es probable que tengan cierta responsabilidad por haber cambiado alguna vez la corriente de la historia.

En Troya los griegos jugaban a los dados así como lo hacía Marco Antonio en Alejandría. Italia y Francia contagiaron a la Europa medieval su gusto por los dados, gusto que ha continuado ininterrumpidamente hasta nuestros días. En este capítulo del libro, investigaremos la manera de utilizar el generador de números aleatorios del ordenador en lugar de los dados físicos. Sin embargo, el lector no tardará en descubrir que los dados no pierden su atractivo por el hecho de existir sólo en el programa de su ordenador más que en una forma física evidente.

Jugar contra el ordenador en vez de contra otro ser humano cambia en cierto modo el «sentimiento» ligado a los dados. ¡El no tener que pagar cuando se pierde es una de las grandes ventajas!

Hay una gran variedad de juegos de dados diversos, todos ellos con diferentes, y a veces, intrigantes nombres.

Los juegos de dados que se incluyen en este capítulo son:

OJOS DE SERPIENTES (SNAKES EYES)
UNO Y VEINTE (ONE-AND-TWENTY)
SIETE/ONCE (SEVEN/ELEVEN)
POR ENCIMA Y POR DEBAJO (OVER'N'UNDER)
SIN SUDOR (NO SWEAT) y
CHEMIN DE ORDENADOR (CHEMIN DE COMPUTER).

CHEMIN DE ORDENADOR

Este juego está basado, como estoy seguro de que ya se habrá dado cuenta el lector, en «Chemin de Fer». En éste, el jugador humano y el ordenador lanzan por turno cinco dados y van sumando los puntos de cada vez. El jugador tiene que conseguir un total de puntos más alto que el del ordenador.

Sin embargo, este juego no es tan simple como parece. Todos los dados que caigan en cinco o en dos han de volverse a tirar y el total es sólo el último número del resultado (es decir, un total de 27 se cuenta como 7 y un total de 13 como 3).

Hay tres totales especiales: 7 (un Natural), 8 (La Petite) y 9 (La Grande). Los nombres sólo se adjudican cuando el total relevante se consigue a la primera tirada (por ejemplo, los dados que cayeron en 5 ó 2 no se han vuelto a tirar). El jugador humano (y el ordenador) siempre insisten en una tirada de 7, 8 ó 9 con el primer lanzamiento de los cinco dados.

Si se fija en el listado del programa, verá que después de inicializar o sembrar el generador de números aleatorios en la línea 20, de borrar la pantalla y de inicializar en cero el contador del programa (JUEGO) y las variables del marcador del jugador humano (P1) y las del marcador del ordenador (B1), el programa va a la línea 330, en donde comienza el juego propiamente dicho.

Como muchos otros programas de este libro, hay una rutina con bucle de retardo al final del listado, a la cual se llama varias veces durante la partida para mejorar la velocidad del juego y para darle al jugador humano la posibilidad de leer en la pantalla lo que está sucediendo antes de que el programa se lance a correr, de borrar la pantalla y de dejarle bastante inseguro de todo lo que acaba de suceder. En

este programa hay dos bucles, uno más largo que el otro y que además imprime un par de líneas blancas antes de volver al programa principal.

La línea 330 llama un retraso y después borra la pantalla antes de aumentar el contador JUEGO en la línea siguiente. Hacia la línea 350 se le dice al jugador humano de qué juego se trata y después el ordenador anuncia que, como él va a adoptar el papel de la banca y ésta siempre abre el juego, será él quien tire primero. Se traba entonces la subrutina de tirar los dados que comienza en la línea 70. Verá que el bucle contador G se usa (línea 80) para ajustar la variable A a un valor escogido aleatoriamente entre uno y seis de cada vez por medio del bucle. Si el dado cae en cinco o en dos (línea 90) se le vuelve a ajustar en cero, antes de que la línea siguiente imprima el valor.

La ejecución total se mide sobre una variable D y luego los dígitos más importantes son divididos a partir de ésta (por ejemplo, 27 se reduce primero a 17 y después a 7) hacia la línea 150. El total cambiante se imprime en la pantalla de cada vez. Después de un retraso el ordenador imprime "el total de la primera tirada es..." y después verifica (líneas 180, 190 y 200), si se ha dado una de las tiradas especiales.

Si ha sido así, se imprime un mensaje a este efecto. Si no ha sucedido nada de esto, el ordenador llega a la línea 210 donde ve si hay que volver a lanzar algún dado (C se incrementa con uno cada vez que sale un 2 o un 5, ver línea 90). Si no hay que volver a tirar ningún dado, el ordenador continúa hasta la línea 410 e imprime "Así que mi total final es...". Sin embargo, si C no es igual a cero, el ordenador atraviesa otro bucle (240 a 300) para volver a tirar esos dados. Otra vez aquí se rechazan los dados que caigan en dos o en cinco (línea 280).

Una vez que el ordenador ha tirado, le toca tirar al jugador humano. En realidad es bastante poco lo que éste tiene que hacer a estas alturas del juego. Una vez que ha presionado RETURN (líneas 440 y 450), el ordenador lo hará todo por él, utilizando la misma rutina que al principio del programa, para tirar los dados en su lugar y aumentando su total al mismo tiempo.

Cuando ya han tirado los dos, el ordenador decide si ha sido él quien ha ganado al jugador humano o a la inversa o si han quedado empatados. El jugador que gane más juegos de los nueve que componen una partida es el ganador (y los juegos empatados no cuentan).

En la línea 670 se le dice "los totales hasta ahora son..." y —si todavía no se han jugado nueve juegos— el ordenador hace algún comentario del tipo de "Parece que voy ganando" o "Parece que tú llevas ventaja", antes de volver al programa principal.

Una vez que el ordenador ha detectado que se han jugado nueve juegos (hacia la línea 700), entra en acción la rutina que comienza en la línea 750. El ordenador le dice al jugador humano "Bueno, muchacho, parece que hemos llegado al final del juego" y después determina quién ha sido el ganador total de la partida.

El jugador humano se lo pasará muy bien jugando a este juego. El «Baccara», del que se deriva «Chemin de Fer», es un juego italiano que se introdujo en Francia hacia 1490 durante el reinado de Carlos VIII.

Este es el aspecto que tiene en acción:

```
>>>> Este es el juego 1 <<<<

*******************************************
Ahora, yo hago de banquero...
*******************************************
 1  4  2  4  5
 9


El total en la primera vuelta es 9
y es La Grande...


Asi que mi total final es 9


*******************************************
Aprieta RETURN para tirar tus dados,
*******************************************
```

```
 1  5  2  4  3
 8

El total en la primera vuelta es 8
y es La petite...

Asi que tu total final es 8

Computadora   Humano

 9             8

$$$
```

```
Asi que tu total final es 9

Computadora   Humano

 6             9

$$$
```

Y este es el listado del programa:

```
 10 REM Chemin de ordenador
 20 RANDOMIZE
 30 CLS
 40 JUEGO = 0
 50 B1 = 0:P1 = 0
 60 GOTO 330
 70 D=0:C=0
 80 FOR G=1 TO 5:A=INT(RND(-1)*6)+1
 90 IF A=2 OR A=5 THEN C=C+1
100 PRINTA;
110 GOSUB 930
120 IF A=2 OR A=5 THEN A=0
130 D=D+A:NEXT G
140 PRINT:PRINTD;
150 IF D>9 THEN D= D-10:PRINT:PRINTD:GOTO150
160 GOSUB 900
170 PRINT:PRINT"El total en la primera vuelta es
"D
180 IF D= 9 THEN PRINT"y es La Grande...":RETURN
190 IF D=8 THEN PRINT "y es La petite...":RETURN
200 IF D= 7THEN PRINT"y es un Natural...":RETURN
210 IF C=0 THEN RETURN
220 GOSUB 930
230 PRINT:PRINTC"debe echarse otra vez"
240 FOR A=1 TO C
250 GOSUB 930
260 E=INT(RND(-1)*6)+1
270 PRINTE;
280 IF E= 2 OR E=5 THEN E=0
290 D=D+E
300 NEXT A
310 IF D>9 THEN D=D-10:PRINTD;:GOTO 310
320 RETURN
330 GOSUB 930:CLS
340 JUEGO=JUEGO+1
350 PRINT:PRINT">>>> Este es el juego"JUEGO"<<<<
":PRINT
360 PRINT"*************************************
******"
370 PRINT"Ahora, yo hago de banquero..."
```

```
  380 PRINT"*************************************
******"
  390 GOSUB 70
  400 GOSUB 900
  410 PRINT:PRINT"Asi que mi total final es"D
  420 GOSUB 900
  430 PRINT"*************************************
******"
  440 INPUT"Aprieta RETURN para tirar tus dados, "
,A$
  450 PRINT"*************************************
******"
  460 J=D
  470 GOSUB 930
  480 GOSUB 70
  490 PRINT:PRINT"Asi que tu total final es"D
  500 PRINT:PRINT
  510 GOSUB 930
  520 PRINT"Computadora","Humano"
  530 GOSUB 900
  540 PRINTJ,D
  550 GOSUB 930:PRINT
  560 PRINT"$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$$$$$$$$$$$$$$$$"
  570 PRINT TAB(5);
  580 IF J= D THEN PRINT "Hemos empatado":GOTO620
  590 IF J>D THEN PRINT"Yo soy el";:B1=B1+1
  600 IF D>J THEN PRINT"Tu eres el";:P1=P1+1
  610 PRINT" vencedor esta vez!"
  620 PRINT"$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$$$$$$$$$$$$$$$$"
  630 PRINT:PRINT
  640 GOSUB 900
  650 PRINT"Los totales hasta ahora en"
  660 PRINT"'Chemin de Ordenador' son"
  670 PRINTB1"para mi, y"P1"para ti..."
  680 PRINT:PRINT
  690 GOSUB 930
  700 IF B1 + P1 = 9 THEN 750
  710 IF B1 > P1 THEN PRINT "Parece que voy ganand
o!"
  720 IF P1 > B1 THEN PRINT "Parece que llevas ven
taja !! "
```

```
730 GOSUB 900
740 GOTO330
750 REM fin del juego
760 PRINT:PRINT
770 PRINT"Bien, muchacho, parece"
780 PRINT"que hemos llegado al final"
790 PRINT"del juego...con un total"
800 PRINT"de nueve tiradas..."
810 GOSUB 900
820 IF P1 > B1 THEN PRINT "Y por una vez, eres t
u":PRINT"el que me ha vencido!":GOTO 840
830 IF B1 > P1 THEN PRINT "Y una vez mas, la tod
opoderosa maquina":PRINT"prueba que es la":PRINT"s
uprema campeona!!!"
840 GOSUB 900
850 PRINT"Gracias por el juego, muchacho"
860 PRINT"debemos repetirlo otra vez,"
870 PRINT"cuando veas que te sientes"
880 PRINT"afortunado....................."
890 END
900 FOR Z=1 TO 1000:NEXT Z
910 PRINT:PRINT
920 RETURN
930 FOR Z= 1 TO 500:NEXT Z:RETURN
```

SIETE/ONCE

Es éste otro divertido juego de dados. En SIETE/ONCE el objetivo que han de alcanzar los jugadores cambia en cada partida. Está basado en otro juego llamado CRAPS y los jugadores han de conseguir en cada baza o jugada el número que se ha fijado para esa jugada (más abajo veremos cómo se fija).

Craps es la versión americana de un antiguo juego de dados inglés llamado Hazard. En SIETE/ONCE se ha suprimido el complejo sistema de apuestas, dejando sólo la parte más divertida que es el juego en sí mismo. En vez de andarse preocupando por computerizar las apuestas desiguales, el jugador puede concentrarse en pedir al cielo que le salgan buenos dados.

En este juego el jugador juega contra sí mismo. El ordenador lleva la cuenta de las bazas que va ganando o perdiendo el jugador humano, de modo que éste sabe en cada momento del juego cómo lo está haciendo.

Se tiran dos dados al mismo tiempo. Si al jugador le sale un 2, 3, ó 12 en la primera tirada, pierde esa baza. Conseguir un 7 o un 11 en la primera tirada es lo mejor que le puede suceder a un jugador ya que ganará esa baza sólo con haber tirado esa vez (por esto el juego se llama SIETE/ONCE).

Cualquier otro número que le salga en la primera tirada se convierte en el número que ha de alcanzar el jugador en la siguiente baza, antes de que le salga un 7 o un 11. Si consigue que le vuelva a salir el mismo número en la siguiente tirada, gana esa baza. Sin embargo, si le vuelve a salir un 7 o un 11 antes de haberlo conseguido, pierde la baza.

Estas son algunas instantáneas del juego en acción:

```
Las puntuaciones hasta ahora:
Victorias: 0    Derrotas: 0
Tienes $ 100
```

```
Esta es la vuelta No 1

Aprieta 'T' para tirar los dados
Has sacado un 2
y 3

En 7/11 has conseguido 5
Tu objetivo es el numero 5

Aprieta 'T' para tirar los dados
```

```
Las puntuaciones hasta ahora:
Victorias: 1    Derrotas: 0
Tienes $ 115

Esta es la vuelta No 2

Aprieta 'T' para tirar los dados
```

Y este es el listado de SIETE/ONCE:

```
 10 REM 7/11
 20 B$ = "En 7/11 has conseguido"
 30 J=0:V=0:D=0:DI=105
 40 CLS
 50 PRINT:PRINT"Las puntuaciones hasta ahora:"
 60 PRINT"Victorias:"V"   Derrotas:"D
 70 J=J+1
 80 DI=DI-5
 90 PRINT"Tienes $"DI
100 PRINT:PRINT"Esta es la vuelta No"J
110 GOSUB 200
120 IF A=7 OR A= 11 THEN 300
130 IF A=2 OR A= 3 OR A= 12 THEN PRINT "asi que
este es el final de la vuelta":GOTO 320
140 P=A
150 PRINT"Tu objetivo es el numero"P
160 GOSUB 200
170 IF A= P THEN 300
```

```
180 IF A= 7 OR A= 11 THEN 310
190 GOSUB 340:GOTO150
200 N=0
210 PRINT:PRINT"Aprieta 'T' para tirar los dados
"
220 N=N+V:IF INKEY$="" THEN 220
230 C=INT(RND(-1)*6)+1:PRINT"Has sacado un"C
240 GOSUB 340
250 B=INT(RND(-1)*6)+1:PRINT"y"B
260 A=B+C
270 PRINT:PRINT B$;A
280 GOSUB 340
290 RETURN
300 PRINT"Y tu has vencido":V=V+1:DI=DI+20:GOTO
320
310 PRINT"Y tu pierdes":D=D+1
320 GOSUB 340
330 GOTO40
340 FOR T=1 TO 500:NEXT T
350 RETURN
```

SIN SUDOR

El nombre de este juego se deriva de uno de los nombres con que se designaba en antiguo inglés al juego a partir del cual éste se ha desarrollado. Conocido como «Sweatcloth», primitivamente se jugaba con tres dados metidos en un «zueco» de madera. La variante más conocida de este juego puede verse todavía hoy en los casinos americanos en donde se le llama «Birdcage» debido al instrumento que se usa para jugar.

En el «Birdcage» se juega con tres dados metidos en una especie de jaula metálica que puede girar en torno a un eje central. Una vez que se han hecho las apuestas, se gira la jaula. Se supone que con esto se asegura que los dados se agitan como es debido ya que, de hecho, el empleado del casino no llega a tocarlos.

Las apuestas en el Birdcage, y Sin Sudor, son rápidas y sencillas y, aunque el jugador pueda pensar que tiene todas las probabilidades de su lado, pronto aprenderá lo fácil que es llegar a perder hasta la camisa.

El jugador apuesta una cantidad determinada (véanse las líneas 50 y 60) que no puede sobrepasar el caudal total con que cuenta para las apuestas (que está contenido en la variable D, de «dinero», véanse las líneas 30 y 250), y después escoge un número entre el uno y el seis. La cantidad de la apuesta se resta al caudal total y luego se tiran los tres dados. Por cada dado que caiga con el número por el que el jugador ha apostado, éste recibe una cantidad igual a la apostada.

Como verá, esto significa que para hacer algo de dinero, el jugador ha de conseguir que le salgan dos dados, por lo menos, con el número por el que ha apostado; en el caso de que le saliera sólo uno, se quedaría igual. La necesidad de tener que sacar dos veces el número escogido es lo que le da al casino una ventaja tan grande en este juego.

Después de inicializar o sembrar el generador de números aleatorios en la línea 20 y de ajustar en 30 la variable del dinero (D) de que dispone en total el jugador en la línea 30, el ordenador va a la subrutina que comienza en la línea 240 y que imprime:

```
*******************************
''tu capital asciende a 30 dólares''
*******************************
```

Tras esto se inicia el juego propiamente dicho.

En la línea 50 se le pregunta al jugador cuánto quiere apostar y se asigna a su elección la variable A. Esta se compara con el caudal con que cuenta en total el jugador (en la línea 60) para ver si está intentando apostar más de lo que en realidad tiene. Habiendo sobrepasado este obstáculo, la línea 70 se encarga de substraer al caudal la cantidad de esta apuesta y después el ordenador le pide que introduzca los números por los que apuesta.

El bucle C, que va de las líneas 100 a la 180, tira los dados y compara cada tirada con el número por el que el jugador ha apostado, diciéndole, después de cada tirada, lo que ha sucedido. Las ganancias del jugador (si llegara a haberlas) se añaden también en este bucle.

Después de cada jugada, el dinero del jugador se compara a 250 dólares (de haber ganado más de 250 dólares, no podrá seguir jugando por haber sobrepasado los límites de la casa) y después con cero (a los que no les queda más dinero no se les permite continuar jugando). Pero si todavía no ha ganado más de 250 dólares ni le queda menos de cero, se le permite continuar.

A continuación, una muestra de SIN SUDOR:

```
  Tu capital asciende a $ 30
****************************************
Cuanto te gustaria apostar?5


A que numero estas apostando?4

El dado 1 ha dado 2
****************************************
  Tu capital asciende a $ 25
****************************************
```

```
El dado 2 ha dado 3
****************************************
  Tu capital asciende a $ 25
****************************************


El dado 3 ha dado 3
****************************************
  Tu capital asciende a $ 25
****************************************



****************************************
Cuanto te gustaria apostar?25


A que numero estas apostando?3

El dado 1 ha dado 3
asi que ganas $ 25
****************************************
  Tu capital asciende a $ 55
****************************************


El dado 2 ha dado 5
****************************************
  Tu capital asciende a $ 55
****************************************

El dado 3 ha dado 2
****************************************
  Tu capital asciende a $ 55
****************************************
```

```
A que numero estas apostando?6

El dado 1 ha dado 1
****************************************
  Tu capital asciende a $ 0
****************************************

El dado 2 ha dado 3
****************************************
  Tu capital asciende a $ 0
****************************************

El dado 3 ha dado 5
****************************************
  Tu capital asciende a $ 0
****************************************
El juego ha terminado, porque no te queda dinero!!
El juego ha terminado, porque no te queda dinero!!
El juego ha terminado, porque no te queda dinero!!
El juego ha terminado, porque no te queda dinero!!
El juego ha terminado, porque no te queda dinero!!
El juego ha terminado, porque no te queda dinero!!
```

Y este es el listado para que se eche unas partidas con el ordenador:

```
10 REM Sin sudor
20 RANDOMIZE
30 D=30
40 CLS :GOSUB 230
50 INPUT"Cuanto te gustaria apostar";A
60 IF A>D THEN 50
70 D=D-A:PRINT:PRINT
80 INPUT"A que numero estas apostando";B
90 IF B<1 OR B> 6 THEN 80
```

```
100 FOR C= 1 TO 3
110 G=0
120 GOSUB 270
130 DA=INT(RND(-1)*6)+1
140 PRINT:PRINT"El dado"C"ha dado"DA
150 IF DA = B THEN G=A:PRINT"asi que ganas $"G:D
=D+G
160 GOSUB 230
170 NEXT C
180 GOSUB 270:GOSUB 270
190 IF D> 250 THEN 300
200 IF D> 0 THEN 40
210 PRINT"El juego ha terminado, porque no te qu
eda dinero!!"
220 GOTO 210
230 PRINT"*************************************
**"
240 PRINT"  Tu capital asciende a $"D
250 PRINT"*************************************
**"
260 RETURN
270 FOR P=1 TO 1000
280 NEXT P
290 RETURN
300 FOR J= 1TO30
310 PRINT"Has alcanzado los $250!!!"
320 PRINTTAB
330 NEXT J
```

UNO Y VEINTE

Continuamos ahora esta sección dedicada a los dados con UNO Y VEINTE que, como se había dado cuenta nada más leer su nombre en la introducción a este apítulo, es una versión en dados de un juego de cartas conocido con el nombre de «Pontoon».

Este juego es bastante simple, pero requiere un tanto de frialdad por parte del jugador, así como cierta habilidad para adivinar qué números serán los que salgan en la siguiente tirada. En este juego el jugador humano juega contra el ordenador.

El jugador humano siempre abre el juego. Puede tirar los dados cuantas veces quiera intentando conseguir un total lo más cercano posible a 21, pero sin pasarse. Puede dejar de tirar en el momento que quiera sumándose los puntos que ha conseguido.

Si se pasa de 21, el ordenador gana esa baza automáticamente. Una partida consta de cinco bazas o jugadas y el ganador de más jugadas es el que gana la partida.

Después de inicializar algunas variables (entre las que se incluyen HS para el marcador del jugador humano y CS para el marcador del ordenador) en las líneas 20 y 30, de borrar la pantalla en la línea 40, el ordenador pedirá al jugador humano que «Presione 'R' para ''tirar'' y 'S' para ''pasar''». La palabra «pasar» o «quedarse» significa que al jugador le parece bien el total que ha conseguido con las veces que ha tirado y no quiere tirar más, dándole al computador la posibilidad de tirar e intentar ganarle. El jugador aprenderá en seguida que la velocidad con la que introduzca 'R' o 'S' puede afectar al número que salga al tirar, ya que el espacio de tiempo comprendido entre la aparición de «Aprieta la 'T' para tirar...» y el momento en que el jugador humano

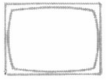
cumple la orden, presionando 'T' o 'P', se usa para crear un número (N) que —junto con el total que el jugador tenga en ese momento— se utiliza para inicializar el generador de números aleatorios para la siguiente tirada.

Las líneas comprendidas entre la 50 y la 170 controlan la tirada de dados del jugador humano y el ordenador sólo expulsa este ciclo cuando éste decide «pasar» (o «quedarse»). La lógica total del juego del ordenador está contenida en la línea 180 que determina si le merece la pena arriesgarse para intentar sobrepasar el total del jugador humano. Si decide tirar, las líneas comprendidas entre la 190 y la 250 controlarán la tirada.

Una vez que el ordenador ha decidido que ya ha tenido bastante, usa las líneas 260, 280 y 290 para determinar quién ha ganado. Si el jugador humano y el ordenador tienen los mismos puntos, o los dos sobrepasan 21, en ese caso, esa mano o jugada no cuenta (en la línea 320 se le indica) y no se aumenta ni el CS ni el HS (el marcador del ordenador y del jugador humano, respectivamente). La línea 280 señala una victoria del ordenador añadiendo uno al CS e imprimiendo "yo gano", mientras que la siguiente línea añade uno a HS e imprime "tú ganas".

Como en el caso de «Chemin de Ordenador», en el que ganaba la partida el jugador que consiguiera ganar más jugadas, bazas o manos de las nueve que componen el juego, en UNO Y VEINTE gana el que consiga más victorias de las cinco jugadas de que consta el juego. La línea 390 mira si ya ha sucedido esto por parte de alguno de los jugadores y si ve que sí ha sucedido, envía el programa a la línea 450 en la que se imprime (junto con algún comentario sarcástico) el resultado total de la partida. Si todavía no se han jugado cinco manos, la línea 400 le dice al jugador humano "Aguarda la siguiente vuelta..." y después de una breve pausa (línea 430), la siguiente jugada le toca al jugador humano.

Echémosle ahora un vistazo a una parte del programa en acción:



```
tu total es 18

Yo saque un 3
asi que mi total es 3
```

```
Yo saque un 3
asi que mi total es 6

Yo saque un 3
asi que mi total es 9

Yo saque un 5
asi que mi total es 14

Yo saque un 6
asi que mi total es 20

************************************************************

             Yo gano!!

Yo saque un 5
asi que mi total es 13

Yo saque un 3
asi que mi total es 16

Yo saque un 5
asi que mi total es 21
Esta vuelta esta empatada...no puntua

************************************************************

Despues de esta vuelta, la puntuacion del juego es
             Tu: 1 , y yo: 1

************************************************************


Aguarda la siguiente vuelta...
```

```
Este es el final del juego

Puntuaciones finales:
                Tu: 1
                Yo: 0

Y asombrosamente, un simple humano ha vencido a la maquina!
```

Este listado le permitirá al lector retar (y con un poco de suerte ganar) a su ordenador en UNO Y VEINTE:

```
   10 REM Uno-y-veinte
   20 HS=0:CS=0
   30 H=0:C=0:N=0
   40 CLS
   50 PRINT"Aprieta la 'T' para tirar, 'P' para pa
sar"
   60 A$=INKEY$
   70 N=N+1
   80 IF A$<>"t" AND A$<> "T" AND A$<>"p" AND A$<>
"P" THEN 60
   90 IF A$="P" OR A$="p" THEN CLS:PRINT"tu total
es"H:GOTO180
  100 RANDOMIZE
  110 GOSUB 430
  120 R=INT(RND(-1)*6)+1
  130 PRINT:PRINT"Has sacado un"R
  140 H=H+R
  150 PRINT"asi que tu total es"H
  160 GOSUB 430
  170 PRINT:GOTO50
  180 IF C> H AND C<22 OR C> 21 OR H> 21 OR H=21 A
ND C= 21 THEN 260
  190 R= INT(RND(-1)*6)+1
  200 GOSUB 430
  210 PRINT:PRINT"Yo saque un"R
  220 C=C+R
  230 PRINT"asi que mi total es"C
```

```
240 GOSUB 430
250 GOTO180
260 IF H=C OR H>21 AND C>21 THEN 320
270 GOSUB 510
280 IF (C>H OR H>21) AND C<22 THEN PRINT ,"Yo ga
no!!";:CS=CS+1
290 IF (C<H OR C>21) AND C<22 THEN PRINT ,"Tu ga
nas!!":HS=HS+1
300 PRINT
310 GOTO330
320 PRINT"Esta vuelta esta empatada...no puntua"
330 GOSUB 430
340 GOSUB 510
350 PRINT"Despues de esta vuelta, la puntuacion
del juego es"
360 GOSUB 430
370 PRINT,"Tu:"HS", y yo:"CS
380 GOSUB 510
390 IF CS+ HS = 5 THEN GOTO 450
400 PRINT:PRINT"Aguarda la siguiente vuelta..."
410 GOSUB 430:GOSUB 430
420 CLS:GOTO30
430 FOR O=1 TO 900:NEXT O
440 RETURN
450 PRINT:PRINT"Este es el final del juego"
460 PRINT:PRINT"Puntuaciones finales:"
470 PRINT,"Tu:"HS
480 PRINT,"Yo:"CS:PRINT
490 IF CS>HS THEN PRINT"Y te he vencido, humanoi
de":END
500 PRINT"Y asombrosamente, un simple humano ha
vencido a la maquina!":END
510 PRINT:PRINT"*********************************
*********************************":PRINT
520 RETURN
```

OJOS DE SERPIENTES

Este juego exige cierta frialdad. El jugador humano y el ordenador tiran por turno un par de dados. El jugador suma los puntos obtenidos en esta baza y después los añade a su marcador.

El jugador puede tirar los dados cuantas veces quiera, pero si le sale un siete, pierde automáticamente. Así pues, como se puede ver, el programa exige que el jugador decida rápidamente si debe tener cuidado con lo que se arriesga a perder esa jugada por no alcanzar los tantos suficientes o si, por el contrario, debe ser ambicioso y debe seguir tirando para conseguir más tantos a riesgo de que le salga un siete.

Como en el juego anterior, el tiempo que tarda el jugador humano en presionar RETURN para echar los dados, tiene cierto efecto sobre los tantos que consiga. Se puede ver en las líneas comprendidas entre la 120 y la 150 que la variable N (que se ajusta en cero en la línea 120) se incrementa con uno cada vez en torno al bucle de las líneas 140/150, hasta que el jugador presiona las teclas "s", "S", "t", "T".

Si el jugador presiona "t" o "T" para indicar que desea echar los dados, el ordenador va a la subrutina que empieza en la línea 370 en la que se borra la pantalla (380) y después se tiran los dados (X y Y). El bucle Q se cubre 20 veces, disminuyendo la velocidad gradualmente según es atravesado (por el bucle interno, en la línea 430), hasta que finalmente —con la línea 420— se imprimen los resultados de las tiradas.

La variable Z se ajusta igual al total de las dos tiradas (en la línea 480) antes de que el ordenador vaya a la línea 700 para el retraso y para imprimir una línea a lo ancho de la pantalla, tras lo cual vuelve al inicio del programa en la línea 180, la siguiente a la que envía el

programa a la rutina de «tirar los dados». Aquí el ordenador comprueba que el total no es 7 (línea 180) y si no lo es, añade el resultado de la última tirada a la cuenta del jugador humano.

Sin embargo, si el jugador humano señala por medio de INKEY$ que «se queda» o «para», la acción va hasta la rutina que comienza en la línea 220. Después de imprimir "Un momento", el ordenador va hasta el bucle de retardo y la rutina de impresión de línea y después usa la misma rutina que el jugador humano para tirar los dados y —al volver de la subrutina— comprueba que el total no es 7, y si no lo es, añade un nuevo total a su marcador. El ordenador tiene un modo muy sencillo de decidir si vuelve a tirar... si tiene menos puntos que el jugador humano, corre el riesgo.

Verá que es relativamente sencillo programar juegos de este tipo. Una vez que se han calculado las rutinas mecánicas que llevan a cabo tareas tales como tirar los dados y aumentar el marcador del jugador humano, no es muy difícil calcular una rutina que permita al ordenador usar las mismas rutinas mecánicas. La mayoría de los juegos de dados no requieren mucha inteligencia y su estrategia puede reducirse a menudo a un par de instrucciones IF/THEN. Lea algún libro sobre juegos de dados —como el excelente *Dice Games Old and New* de William E. Tredd— si quiere sacar ideas que pueda convertir en programas y para aprender a jugar lo mejor posible. Son estas ideas, cuya conversión en «algoritmos inteligentes» no debería resultarle especialmente dificultosa, las que harán que su ordenador juegue razonablemente bien contra usted.

Y por último, si como en el caso de UNO Y VEINTE, el jugador humano estipula que ha de ser él quien tire primero, el ordenador sabrá exactamente cuál es su objetivo (puesto que ya conoce los puntos del jugador humano) y, por tanto, llevará ventaja a la hora de tirar sus dados, lo que ayuda a superar la estupidez inherente a la máquina.

Este es un ejemplo típico de la ejecución del juego:

```
Tu total es 0

----------------------------------------

Aprieta 'T' para tirar
o 'S' para salir.
```

```
---------------------------------------

Dado uno: 6      Dado dos: 6

Tu total es 12


Tu total es 12

---------------------------------------

Aprieta 'T' para tirar
o 'S' para salir.


---------------------------------------

Dado uno: 2      Dado dos: 6

Tu total es 20


Tu total es 20

---------------------------------------

Aprieta 'T' para tirar
o 'S' para salir.

Un momento
```

```
---------------------------------------

Dado uno: 5      Dado dos: 6

Mi total es 11
Tu total es 22


Un momento

---------------------------------------

Dado uno: 2      Dado dos: 1

Mi total es 14
Tu total es 22


Un momento

---------------------------------------

Dado uno: 6      Dado dos: 4

Mi total es 24
Tu total es 22

Yo gano!

La puntuacion es:
              Tu: 0
              Yo: 1
---------------------------------------

Y voy ganando...
```

Y este es el listado del juego.

```
 10 REM Ojos de serpientes
 20 CLS
 30 H=0:CZ=0
 40 HS=0:CS=0
 50 PRINT"Un momento, por favor..."
 60 GOSUB 700
 70 CLS:PRINT:PRINT
 80 PRINT"Tu total es"HS:PRINT
 90 GOSUB 710
100 PRINT"Aprieta 'T' para tirar"
110 PRINT"o 'S' para salir."
120 N=0
130 W$=INKEY$
140 N=N+1
150 IF W$<> "s" AND W$<>"S" AND W$ <> "T" AND W$
<>"t" THEN RANDOMIZE:GOTO130
160 IF W$="S" OR W$="s" THEN 220
170 GOSUB 370
180 IF Z= 7 THEN 350
190 HS=HS+Z
200 PRINT:PRINT"Tu total es"HS
210 GOTO60
220 PRINT:PRINT:PRINT"Un momento"
230 GOSUB 700
240 GOSUB 370
250 IF Z=7 THEN 330
260 CS=CS+Z
270 PRINT:PRINT"Mi total es"CS
280 PRINT"Tu total es"HS
290 IF CS< HS THEN 220
300 IF CS=HS THEN PRINT"Es ahogado!"
310 IF CS>HS THEN 350
320 GOTO510
330 PRINT:PRINT"Tu ganas!":H=H+1
340 GOTO510
350 PRINT:PRINT"Yo gano!":CZ=CZ+1
360 GOTO510
370 REM Tira los dados
380 CLS
```

```
390 FOR Q=1 TO 20
400 X= INT(RND(-N)*6)+1
410 Y= INT(RND(-N)*6)+1
420 PRINT:PRINT"Dado uno:"X      Dado dos:"Y
430 FOR P=1 TO 3*Q:NEXT P
440 NEXT Q
450 CLS
460 GOSUB 710
470 PRINT"Dado uno:"X      Dado dos:"Y
480 Z=X+Y
490 GOSUB 700
500 RETURN
510 PRINT:PRINT"La puntuacion es:"
520 PRINT,"Tu:"H
530 PRINT,"Yo:"CZ
540 IF CZ+H = 9 THEN 580
550 GOSUB 710
560 IF H> CZ THEN PRINT"Vas ganando!"
570 IF CZ>H THEN PRINT"Y voy ganando..."
580 GOSUB 700
590 CLS
600 IF CZ+H=9 THEN 620
610 GOTO40
620 PRINT:PRINT:PRINT
630 PRINT"Bueno, muchacho, este es el final"
640 PRINT"     del juego..."
650 PRINT:PRINT:PRINT
660 PRINT"Tu puntuacion final fue"H
670 PRINT"Y la mia fue"CZ:PRINT
680 IF H<CZ THEN PRINT"Soy el vencedor!!":END
690 PRINT"Tu eres el vencedor!!":END
700 FOR P=1 TO 1000:NEXT P:RETURN
710 PRINT"------------------------------------------------------------
":PRINT:RETURN
```

POR ENCIMA Y POR DEBAJO

Este juego es una adaptación del juego de cartas y dados generalmente conocido como «las siete y media». En este juego, el jugador apuesta sobre la posibilidad de que el total de puntos conseguidos al tirar los dados (se juega con dos) sea:

— menos de siete
— siete; o
— más de siete

Este vuelve a ser otro juego en el que —a primera vista— parece que se le ofrecen al jugador buenas probabilidades. Sin embargo, si jugara una partida interminable con un par de dados perfectos encontraría que las derrotas sobrepasan a las victorias en casi un 17 por 100.

La estrucura del programa no es difícil de seguir. Después de inicializar el generador de números aleatorios en la línea 20 y de borrar la pantalla en la 30, el ordenador fija en 30 dólares la cantidad total de que dispone el jugador humano para las apuestas (variable que se llama aquí DI de «dinero»). La línea 50 envía la acción a la subrutina que comienza en la línea 450 en la que se imprime esa cantidad: "Tienes 30 dólares", antes de volver a la línea 60 en donde se llama a una subrutina de retraso (líneas 470, 480 y 490).

Las líneas comprendidas entre la 70 y la 120 le piden que haga su apuesta, dándole la clave para introducirla (presione A si apuesta a que sale menos de siete; B para siete y C en el caso de que apueste a un total superior a siete). Las líneas 130 y 140 leen las claves utilizando INKEY$ y rechazando (línea 140) toda información que no sea A, B o C.

Habiendo recibido una elección válida, las líneas comprendidas entre la 150 y la 180 imprimen la apuesta y luego la línea 200 le pide que introduzca la cantidad de dinero que quiere arriesgar en ésta. Por supuesto, no puede apostar más dinero del que tiene (o por lo menos, no se puede en este juego) de modo que la línea 220 comprueba la apuesta del jugador (A) en relación con su dinero (DI) y si éste no tiene suficiente, imprime: "No tienes tanto".

La siguiente sección tira los dados asignando valores aleatorios entre uno y seis a las variables B y C y sumándolos para producir un total D en la línea 280. En la línea 300 se le dice al jugador cuál ha sido el resultado total de la tirada.

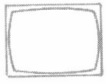
EL RESULTADO

Viene determinado por la rutina que va de la línea 310 a la 350 en donde se supone primeramente una pérdida (línea 310, con la variable G, de «ganar», asignada a la parte negativa de la apuesta del jugador). Esto se modifica —si fuera necesario— en la cantidad correcta en el caso de que se gane, si es que ha sido así. Se le pagará al jugador cuatro veces el dinero que ha apostado en el caso de que haya apostado a siete y el resultado de los dados sea éste. En el caso de haber apostado por encima o por debajo de siete y de acertar, el jugador recibirá la misma cantidad de dinero que ha apostado (línea 320 y 330 y 340 respectivamente).

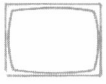
Las líneas 370 y 380 le dicen si ha ganado o perdido y después la línea 420 comprueba si todavía le queda dinero al jugador. Si le queda, se le ofrece echar otra mano. Si no le queda, el jugador es rechazado severamente.

Vea una muestra de cómo funciona el programa:

```
OK, muchacho, es el momento de
hacer tu apuesta..............
```



```
Aprieta 'A' para apostar menos que 7
        'B' para apostar al 7, o
        'C' para appostar sobre 7
```



```
Las apuestas estan:
  A - paga lo mismo
  B - paga 4 a 1
  C - paga lo mismo


Cuanto quieres apostar?10


                El dado uno da 4

                El dado dos da 4



            con lo que el total es 8

Y por lo tanto pierdes $ 10


Ahora tienes $ 20


 OK, muchacho, es el momento de
 hacer tu apuesta..............


 Aprieta 'A' para apostar menos que 7
         'B' para apostar al 7, o
         'C' para appostar sobre 7
```

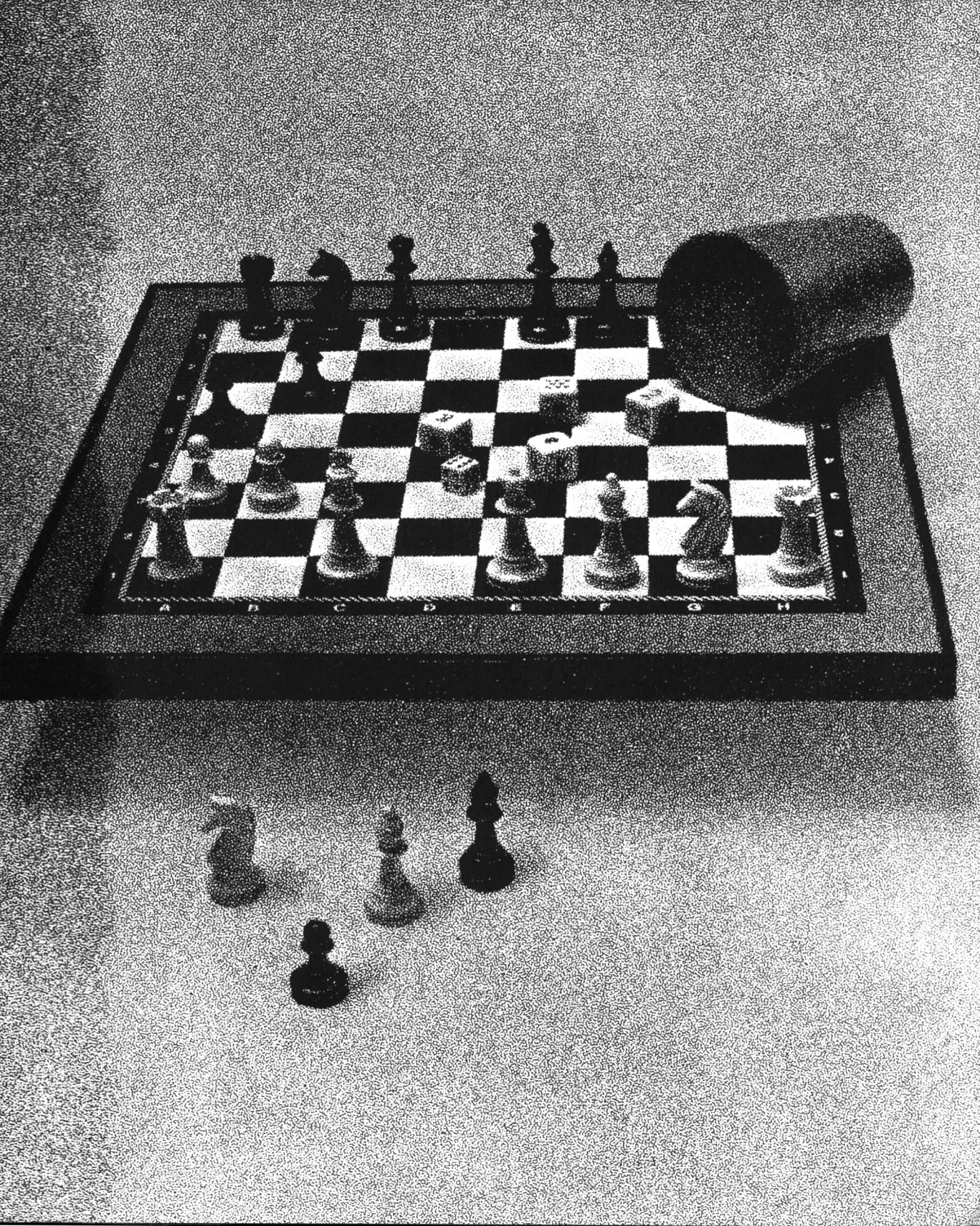
```
Las apuestas estan:
  A - paga lo mismo
  B - paga 4 a 1
  C - paga lo mismo

Cuanto quieres apostar?20

                El dado uno da 3

                El dado dos da 4

            con lo que el total es 7

Y por lo tanto pierdes $ 20


Ahora tienes $ 0


No tienes un duro, muchacho
asi que tengo que cerrar el casino
No hay sitio para los arruinados junto a mi.
```

Y este es el listado:

```
10 REM Por encima y por debajo
20 RANDOMIZE
30 CLS
```

```
 40 DI=30
 50 GOSUB 450
 60 GOSUB 470
 70 PRINT"OK, muchacho, es el momento de"
 80 PRINT"hacer tu apuesta..............."
 90 GOSUB 470
100 PRINT"Aprieta 'A' para apostar menos que 7"
110 PRINT"         'B' para apostar al 7, o"
120 PRINT"         'C' para apostar sobre 7"
130 A$=INKEY$
140 IFA$<>"A" AND A$<>"a" AND A$<>"B" AND A$<>"b
" AND A$ <>"C" AND A$<>"c" THEN 130
150 PRINT:PRINT"Las apuestas estan:"
160 PRINT"  A - paga lo mismo"
170 PRINT"  B - paga 4 a 1"
180 PRINT"  C - paga lo mismo"
190 GOSUB 470
200 INPUT"Cuanto quieres apostar";A
210 GOSUB 470
220 IF A> DI THEN PRINT "No tienes tanto!":PRINT
:GOTO200
230 B=INT(RND(-1)*6)+1
240 PRINT ,"El dado uno da"B
250 GOSUB 470
260 C=INT(RND(-1)*6)+1
270 PRINT,"El dado dos da"C
280 D=C+B
290 GOSUB 470
300 PRINT,"con lo que el total es"D
310 G=-A
320 IF D=7 AND(A$="B" OR A$="b") THEN G=A*4
330 IF D<7 AND(A$="A" OR A$="a") THEN G=A
340 IF D>7 AND(A$="C" OR A$="c") THEN G=A
350 DI = DI + G
360 GOSUB 470
370 IF G>0 THEN PRINT"Acabas de ganar $"G
380 IF G<0 THEN PRINT"Y por lo tanto pierdes $"(
G* -1)
390 GOSUB 470
400 GOSUB 450
410 GOSUB 470
420 IF DI < 1 THEN PRINT"No tienes un duro, much
acho":PRINT"asi que tengo que cerrar el casino":PR
```

```
INT"No hay sitio para los arruinados junto a mi.":
END
  430 CLS
  440 GOTO 50
  450 PRINT:PRINT"Ahora tienes $"DI
  460 RETURN
  470 FOR P=1 TO 1000:NEXT P
  480 PRINT:PRINT
  490 RETURN
```


A B C D E F G H

INTELIGENCIA ARTIFICIAL

La inteligencia artificial es una meta que todavía no se ha alcanzado. Es cierto que se han escrito programas que permiten a los ordenadores dar muestras de un comportamiento que podría calificarse de inteligente, pero la «inteligencia» que exhiben es limitada y efectiva sólo en un dominio muy restringido. Es decir, puede parecer que un ordenador es inteligente, pero sólo lo será si el medio en el que tienen que demostrar esa inteligencia, es un medio limitado.

En este capítulo del libro, se incluyen cuatro programas que harán que su ordenador parezca inteligente y cuya demostración dejará impresionados a quienes no entienden mucho de ordenadores.

Aunque hay quienes afirman (como Carl Sagan *In Defense of Robots*) que los seres humanos no son sino ordenadores bien confeccionados, yo creo que existe un profundo abismo (al menos por el momento) entre nuestra propia percepción de nosotros mismos y la total falta de conciencia que los ordenadores tienen de su propia existencia. A mí me parece que esta percepción de uno mismo es consustancial a, al menos, un aspecto de la inteligencia.

Los ordenadores pueden funcionar de un modo extremadamente «inteligente» en áreas restringidas. Fíjese el lector en esas pequeñísimas máquinas de jugar al ajedrez, por ejemplo. Dentro del dominio limitado de ajedrez puede crearse una sólida apariencia de inteligencia.

Lo mismo sucede en los limitados universos creados por los programas de este capítulo. Uno de ellos es una variación de uno de los programas más conocidos que hayan existido nunca, ELIZA, en el cual el ordenador hace el papel de un psiquiatra y aparentemente es capaz de llevar a cabo una conversación. ELIZA se desarrolló a mediados de los años sesenta y su autor es Joseph Weizenbaum. Intentaba éste producir un programa que caricaturizara a un psiquiatra rogeriano (de la escuela de C. Rogers) y se ha sugerido (véase *You're never alone with a fluffy micro,* de David Tebbutt; MicroScope, enero, 1983, pág. 9) que Weizenbaum no ha dejado de arrepentirse desde entonces de su trabajo en ELIZA. Y no es difícil saber por qué.

En la inmediata euforia que siguió a la publicación del programa, se llegó a afirmar, de un modo bastante a la ligera y sin fundamento alguno, que se estaba ya a un paso de que los ordenadores pudieran reemplazar a los «doctores de la mente». E incluso hubo observadores que fueron todavía más lejos afirmando que el éxito obtenido por el ELIZA abría el camino para la aparición de ordenadores genuinamente inteligentes.

Weizenbaum no daba crédito a sus ojos. Posteriormente escribiría que al principio se quedó totalmente sorprendido —y lo seguía estando— de la reacción del público ante el programa. Escrito en LISP en el MIT, se podía acceder al programa desde cualquiera de las terminales compartidas que había repartidas por el campus. Weizenbaum dice que le interesó en gran medida ver que eran muchas las veces en que se accedía al programa por la noche, como si hubiera estudiantes preocupados que se dedicaran a hablar de sus problemas con el ordenador. Pero eso no era todo. Los tiempos de acceso eran además muy largos. El registro del ordenador demostró que los había que se pasaban más de una hora enganchados al ELIZA.

La misma secretaria de Weizenbaum había trabajado con él durante los seis meses que duró aproximadamente la producción del programa y conocía tan bien como él el funcionamiento de éste. Es decir, como el ordenador explora el discurso de la persona en busca de «palabras clave» (como «dream», «sueño», o «friends», «amigos»), para posteriormente escoger una respuesta adecuada en el banco de respuestas que lleva incorporado. En el caso de que no reconozca ninguna palabra clave, el ordenador puede incorporar otras palabras a partir de las frases del usuario, frases a las que da la vuelta y convierte a su vez en preguntas (de modo que una frase como «el buen tiempo me pone contento», podría aprovecharse ya sea como «¿por qué te

pone contento el buen tiempo?», o como una simple afirmación: «el buen tiempo te pone contento. ¿Qué te sugiere esto?») y además también puede usar un número determinado de otras respuestas («eso es interesante, sigue por ahí»).

Pero a pesar de todo esto y de su familiaridad con el programa, Weizenbaum notó que cada vez que entraba en el despacho cuando su secretaria estaba accediendo al programa, ésta se mostraba un tanto avergonzada y se negaba a enseñarle lo que había salido impreso. Cuando Weizenbaum sugirió que se conectase una impresora en la unidad central del ordenador para recoger las conversaciones que los estudiantes solían tener con éste a altas horas de la madrugada, le miraron horrorizados, como si lo que estuviera sugiriendo fuera algún tipo de espionaje ilegal o escucha telefónica. (Si quiere saber más sobre el trabajo de Weizenbaum, puede que le interese consultar su libro *Computer Power and Human Reason,* W. H. Freeman and Co., San Francisco, 1976.)

Un punto en el que tropezaron todos los primeros investigadores en el campo de la inteligencia artificial fue el hecho de que escribir un programa que emulara en algún aspecto el razonamiento humano (como un impresionante programa de ajedrez producido por Samuels) no les acercaba necesariamente ni a la producción de un compartamiento que pudiera calificarse de inteligente, ni tan siquiera a una comprensión teórica de los procesos de razonamiento y deducción humanos. «El progreso realizado en la producción de un comportamiento inteligente no significa necesariamente que se progrese en la producción de pensamiento» (Norman Whaland, «When Is a Program Inteligent?», publicado en la revista *Creative Computing,* febrero 1981, páginas 44-49).

Hay otra manera de abordar el problema, manera que yo he denominado «si grazna como un pato, es un pato». En el libro *The Turing Criterion. Machine Intelligent Programs for the 16K ZX81* (Charlton y otros, Interface Publications, London, 1982) de cuya edición yo me encargué, contesto a la pregunta «¿Qué es el Turing Criterion?» en los siguientes términos:

> En 1950, el matemático inglés A. M. Turing propuso lo que posteriormente se conocería con el nombre de «criterio Turing» sobre la inteligencia de las máquinas.
>
> Decía éste que si uno se las está viendo con algo situado al final de un cable que podría ser una máquina o un ser humano

y uno no pudiera decir a ciencia cierta —a partir de las respuestas que le vienen por medio del cable— si se trata de una máquina o de un ser humano, «la cosa» al otro lado del cable sería, por definición, inteligente.

Sobre esta base sería posible clasificar muchos, muchos programas de ordenador como inteligentes. Ciertamente, la mayoría de los incluidos en este capítulo del libro lograrían pasar el «test», especialmente si el lector está dispuesto a aceptar la posibilidad de tratarse con ciertos humanos bastante estúpidos.

Ya ha llegado el momento de explorar los mundos de la inteligencia artificial con los siguientes programas:

— Eliza.
— Cerebro electrónico (en el que el ordenador intenta romper el código que el jugador ha creado).
— Proboscídeos (su nombre se deriva del adjetivo que clasifica a ciertos mamíferos, como elefante. En éste el jugador se enfrenta al ordenador en un juego de «concentración»).
— Amanuense (en el que el ordenador acepta la tarea de convertirse en poeta).

ELIZA

Empezaremos nuestra investigación de la inteligencia artificial con una versión simplificada del programa clásico ELIZA sobre el que ya hemos hablado con cierto detenimiento en la introducción al capítulo. Como decíamos, el programa explora el discurso del usuario en busca de palabras que le sirvan para después contestarle, utilizando esas mismas palabras de un modo que parece que está de verdad hablando con él. Una vez que haya jugado con él unas cuantas veces, el usuario aprenderá el modo de conseguir que dé las respuestas más apropiadas.

Lo que sigue es una parte de una «conversación» que mantuve con mi versión de ELIZA:

```
BIENVENIDO A OTRA SESION CON
TU COMPUTADORA SIQUIATRICA, ELIZA

COMO TE LLAMAS?JAVIER


BIEN, HOLA JAVIER

TE ASEGURO QUE ES AGRADABLE TENERTE POR AQUI

OPRIME LA BARRA ESPACIADORA CUANDO ESTES PREPARADO PARA EMPEZAR
```

?TENGO UN GRAVE PROBLEMA
PUEDES PUNTUALIZAR ALGO MAS SOBRE ESO?

?TENGO UN PROBLEMA SOBRE EL SEXO
JAVIER, CREO QUE REALMENTE DEBERIAS DISCUTIR ESO CON UN HUMANO.

?ME GUSTAN LAS COMPUTADORAS
ES BUENO QUE TE GUSTEN LAS COMPUTADORAS

?PUEDES AYUDARME ?
QUIZAS TE GUSTARIA SER CAPAZ DE AYUDARME ?

?CREO QUE NO ME HAS ENTENDIDO
QUE SE SIENTE CREYENDO QUE NO ME HAS ENTENDIDO

?NECESITO UN AMIGO
PORQUE HAS MENCIONADO EL TEMA DE LOS AMIGOS?

?SIEMPRE ME PREOCUPA
PUEDES PENSAR EN UN CASO CONCRETO?

?

?ME SIENTO MAL
JAVIER, PORQUE TE DISCULPAS?

?PORQUE NO PUEDO SEGUIR ASI
QUIZAS NO QUIERES SER CAPAZ DE SEGUIR ASI

?VOY A BEBER
LA MODERACION DEBE PRESIDIR TODOS TUS ACTOS.

?ERES UNA COMPUTADORA ESTUPIDA
QUE TE PARECE QUE SEA UNA COMPUTADORA ESTUPIDA

```
?ME PARECE MAL
NO ESTOY SEGURA DE ENTENDERTE COMPLETAMENTE.

?ME GUSTA TU INGENUIDAD
ES BUENO QUE TE GUSTE TU INGENUIDAD
```

Y este es el listado con el que el ordenador podrá ayudarle a poner un poco de orden en su cabeza:

```
   10 REM Eliza
   20 REM TODAS LAS ENTRADAS EN MAYUSCULAS
   30 REM      SIN SIGNOS DE PUNTUACION
   40 REM        NI COMAS NI ACENTOS
   50 GOSUB 660:REM Inicializacion
   60 PRINT"BIENVENIDO A OTRA SESION CON "
   70 PRINT"TU COMPUTADORA SIQUIATRICA, ELIZA"
   80 PRINT:INPUT"COMO TE LLAMAS";Z$
   90 CLS:PRINT"BIEN, HOLA ";Z$
  100 PRINT:PRINT"TE ASEGURO QUE ES AGRADABLE TENE
RTE POR AQUI"
  110 PRINT:PRINT"OPRIME LA BARRA ESPACIADORA CUAN
DO ESTES PREPARADO PARA EMPEZAR"
  120 N=N+1
  130 IF INKEY$="" THEN 120
  140 CLS:RANDOMIZE
  150 GOSUB 590:REM ACEPTA LA ENTRADA DEL USUARIO
  160 IF RND(-N)<.2 THEN PRINT Z$;", ";
  170 GOSUB 190:REM PROCESA LA ENTRADA, IMPRIME LA
 RESPUESTA
  180 PRINT:GOTO150
  190 REM ENCUENTRA LA PALABRA CENTRAL DE LA FRASE
  200 Z=0
  210 Z=Z+1
  220 D=INSTR(A$,B$(Z))
  230 IF D<>0 THEN 260
  240 IF Z< 35 THEN 210
  250 GOSUB 440:RETURN
  260 REM PROCESA LAS PALABRAS CLAVE
```

```
270 PRINT C$(Z);" ";
280 IF RIGHT$(C$(Z),1) < "A" THEN PRINT:RETURN
290 H=LEN(A$)-(D+LEN(B$(Z)))
300 IF H>0 THEN A$=RIGHT$(A$,H)
310 Z=0
320 Z=Z+1
330 D=INSTR(A$,F$(Z))
340 IF D<>0 THEN 370
350 IF Z<8 THEN 320
360 PRINTA$:RETURN
370 J$=LEFT$(A$,(D-1)) + " " +G$(Z)
380 Z=LEN(A$)-LEN(LEFT$(A$,(D-1)))-LEN(G$(Z))
390 PRINT J$
400 IF Z>2 THEN L$ = RIGHT$(A$,(Z-2)):IF INSTR(K
$,L$) = 0 THEN RETURN
410 IF Z> 2 THEN PRINT RIGHT$(A$,(Z-2))
420 IF Z<2 THEN PRINT
430 RETURN
440 REM RESPUESTAS ALEATORIAS (No hay palabra cl
ave)
450 Z=INT(RND(-N)*11)+1
460 ON Z GOSUB 480,490,500,510,520,530,540,550,5
60,570,580
470 RETURN
480 PRINT"QUE ES LO QUE TE SUGIERE ESO?":RETURN
490 PRINT"¥A VEO...":RETURN
500 PRINT"NO ESTOY SEGURA DE ENTENDERTE COMPLETA
MENTE.":RETURN
510 PRINT"PUEDES PUNTUALIZAR ALGO MAS SOBRE ESO?
":RETURN
520 PRINT"ESO ES MUY INTERESANTE!":RETURN
530 PRINT"ASI SON LAS COSAS...POR FAVOR CONTINUA
...":RETURN
540 PRINT"YA ENTIENDO...":RETURN
550 PRINT"BIEN, BIEN...PROSIGUE.":RETURN
560 PRINT"POR QUE DICES ESO?":RETURN
570 PRINT"POR FAVOR, EXPLICA QUE HAY DETRAS DE E
SA AFIRMACION...":RETURN
580 PRINT"PODRIAS DECIR ESO DE NUEVO, DE UNA MAN
ERA DIFERENTE?":RETURN
590 REM ACEPTA LA  FRASE DEL USUARIO
600 INPUT A$:K$=A$
610 IF LEFT$(A$,5)="ADIOS" THEN PRINT "ADIOS POR
 AHORA.":PRINT"ESPERO VOLVER A VERTE.":END
```

```
620 K=LEN(A$)
630 IF LEFT$(A$,7)="POR QUE" THEN A$=RIGHT$(A$,(
K-7)):K=K-7
640 A$=" "+A$+" "
650 RETURN
660 REM INICIALIZACION
670 CLS:N=0
680 DIM B$(35),C$(35),F$(8),G$(8)
690 REM LLENA B$, C$ CON PALABRAS CLAVE, RESPUES
TAS
700 FOR A=1 TO 35
710 READ B$(A),C$(A)
720 NEXT A
730 RESTORE 1150
740 FOR A=1 TO 8
750 READF$(A):READG$(A)
760 NEXT A
770 RETURN
780 REM PALABRAS CLAVE, RESPUESTAS
790 DATA" PUEDES","QUIZAS TE GUSTARIA SER CAPAZ
DE"
800 DATA" SUELES ","SI, YO SUELO"
810 DATA" PUEDO","QUIZAS NO QUIERES SER CAPAZ DE
"
820 DATA" TU ERES","QUE TE HACE CREER QUE SOY"
830 DATA" ERES","QUE TE PARECE QUE SEA"
840 DATA" YO NO","PORQUE TU NO"
850 DATA" CREO","QUE SE SIENTE CREYENDO"
860 DATA" PORQUE TU NO","PORQUE TE GUSTARIA QUE
YO"
870 DATA" POR QUE NO PUEDO","QUE TE HACE CREER Q
UE DEBERIAS PODER"
880 DATA" ERES","PORQUE TE INTERESA SI SOY O NO"
890 DATA" NO PUEDO","COMO SABES QUE NO PUEDES"
900 DATA" SEXO","CREO QUE REALMENTE DEBERIAS DIS
CUTIR ESO CON UN HUMANO."
910 DATA" ESTOY","CUANTO TIEMPO HAS ESTADO"
920 DATA" YO SOY","POR QUE ME DICES QUE ERES"
930 DATA" QUIERO","PORQUE QUIERES"
940 DATA" QUE","TU QUE CREES?"
950 DATA" COMO","CUAL ES LA RESPUESTA QUE MAS TE
 AGRADARIA?"
960 DATA" QUIEN","CON QUE FRECUENCIA PIENSAS EN
ESAS COSAS?"
```

```
 970 DATA" DONDE","PORQUE ME HAS PREGUNTADO ESO?"
 980 DATA" CUANDO","QUE DIRIA TU MEJOR AMIGO A ES
A PREGUNTA?"
 990 DATA" PORQUE","QUE ES LO QUE REALMENTE QUIER
ES SABER?"
1000 DATA" QUIZAS","NO PARECES ESTAR MUY SEGURO!"
1010 DATA" BEBER","LA MODERACION DEBE PRESIDIR TO
DOS TUS ACTOS."
1020 DATA" SIENTO","PORQUE TE DISCULPAS?"
1030 DATA" SUENO","PORQUE HAS SACADO EL TEMA DE L
OS SUENOS?"
1040 DATA" ME GUSTA","ES BUENO QUE TE GUSTE"
1050 DATA" QUIZAS","NO ESTAS SIENDO UN POCO TENTA
TIVO?"
1060 DATA" NO ","PORQUE ERES TAN NEGATIVO?"
1070 DATA" TUS","PORQUE TE PREOCUPAN MIS"
1080 DATA" SIEMPRE","PUEDES PENSAR EN UN CASO CON
CRETO?"
1090 DATA" PIENSO","ACASO DUDAS"
1100 DATA" SI","PARECES MUY SEGURO. PORQUE?"
1110 DATA" AMIGO","PORQUE HAS MENCIONADO EL TEMA
DE LOS AMIGOS?"
1120 DATA" COMPUTADORA","PORQUE MENCIONAS LAS COM
PUTADORAS?"
1130 DATA" SOY YO","TU ERES"
1140 REM CONJUGACIONES
1150 DATA" YO ","TU"
1160 DATA" ERES ","SOY"
1170 DATA" ERAS ","ERA"
1180 DATA" TI ","MI"
1190 DATA" TU ","MI"
1200 DATA" TENGO ","TIENES"
1210 DATA" ESTOY ","ESTAS"
1220 DATA" MI ","TI"
1230 DATA" SOY YO ","ERES TU"
1240 DATA" ESTOY ","ESTAS"
```

CEREBRO ELECTRONICO

CEREBRO ELECTRONICO es un intento de escribir un programa que funcionara, a la hora de dar solución a un problema, de un modo aparentemente inteligente. El problema que el ordenador intenta solucionar es del tipo de adivinar el código numérico que ha pensado el jugador humano. La información que el jugador vuelve a introducir después de la respuesta del ordenador se da en la forma de «blancos» y «negros». Se da un «blanco» cada vez que el ordenador dice un número correcto, pero no está bien situado en la secuencia del código y un «negro» cuando el ordenador dice un número correcto y debidamente situado dentro del código.

El problema no es tan sencillo como parece ya que el ordenador no sabe con seguridad qué número (o números) ha dado lugar a un blanco (o blancos) y qué número ha dado lugar a un negro (o negros). Escribí dos versiones de este programa. Una en la que el ordenador puede adivinar códigos compuestos por tres números y otra, una modificación de la anterior, en la que el ordenador puede adivinar códigos compuestos de cuatro dígitos. Funciona de un modo muy simple (aunque aplicar la idea relativamente sencilla que sustenta el programa no fue particularmente fácil). Cada vez que se le da un negro a un número de un código, se le da un valor a todos los números de ese código de modo que aparezcan con más frecuencia en los siguientes cálculos. Cuantos más negros tenga un código determinado, más valor se le dará a los números de ese código. A los códigos que consiguen uno o más blancos se les da mucho menos valor. Los números de los códigos que no consiguen ni blancos ni negros dejan de ser tenidos en consideración.

Como dije, la versión de tres números es la base sobre la que se escribió el programa de cuatro números. Introduzca primero la versión de tres números y consérvela así, antes de hacer la conversión necesaria para que funcione como un programa de cuatro números. Mientras que la versión de tres números juega razonablemente bien y razonablemente rápido, el programa de cuatro números hacia el final empieza a renquear un poco e intenga generar códigos que: *a)* reflejan los valores que han producido las contestaciones anteriores y *b)* no son códigos que se hayan sugerido previamente en esa partida.

Aquí tiene el programa intentando adivinar un código de tres números:

```
Intento numero 1
Mi intento es 1  2  3

Cuantos negros?1

 cuantos blancos?0
```

```
Intento numero 2
Mi intento es 4  5  6

Cuantos negros?0

 cuantos blancos?1
```

```
Intento numero 3
Mi intento es 7  8  9

Cuantos negros?1

 cuantos blancos?0
```

```
Intento numero 7
Mi intento es 1  4  9

Cuantos negros?3

He acertado tu codigo: es el  1  4  9
     en solo 7 intentos.
```

Y este es el listado para producir ese programa:

```
   10 REM Cerebro electronico - 3 digitos
   20 GOSUB 510:REM Inicializacion
   30 REM Intenta adivinar
   40 IF PRUEBA <3 THEN FOR Z=1 TO 3:B(Z) = Z+3* P
RUEBA:NEXT Z:GOTO70
   60 GOSUB 280
   70 CLS
   80 PRUEBA = PRUEBA + 1
   90 PRINT:PRINT
  100 PRINT"Intento numero" PRUEBA
  110 PRINT
  120 PRINT"Mi intento es"B(1);B(2);B(3)
  130 PRINT:PRINT
  140 INPUT"Cuantos negros";N
  150 IF N=3 THEN GOTO 600
  160 IF N=2 THEN 190
  170 PRINT:PRINT
  180 INPUT" cuantos blancos";B
  190 IF B+N=3 THEN Q=3:C(1)=B(1):C(2)=B(2):C(3)=B
(3)
  200 IF N+B=0 THEN C(B(1))=0:C(B(2))=0:C(B(3))=0:
GOTO30
  210 IF N>AYUDA THEN FOR Z=1 TO 3:E(Z)=B(Z):NEXTZ
:AYUDA=N
  220 FOR Z=1 TO 9
  230 FOR D=1 TO 3
  240 IF B(D) = C(Z) THEN C(Z) = C(Z) + (B + W)*10
0 + W*10
  250 NEXTD
  260 NEXT Z
  270 GOTO30
  280 REM Coje 3 numeros
  290 FOR Z= 1TO 3
  300 D1=C(INT(RND(-1)*Q)+1)
  310 IF D1 = 0 THEN 300
  320 D2= C(INT(RND(-1)*Q)+1)
  330 IF D2=0 THEN 320
  340 IF INT(D1/10) > INT(D2/10) THEN B(Z) = D1
  350 IF INT(D1/10) < INT(D2/10) THEN B(Z) = D2
```

```
360 IF INT(D1/10) = INT(D2/10) THEN B(Z) = D1
370 IF B(Z)>100 THEN B(Z)= B(Z)-100*INT(B(Z)/100
):GOTO370
380 IF B(Z)>10 THEN B(Z)=B(Z)-10*INT(B(Z)/10):GO
TO 380
390 NEXT Z
400 IF B(1)=B(2) OR B(1)=B(3) OR B(2)=B(3) THEN
290
410 IF AYUDA >0 THEN CUENTA=0:FOR Z=1 TO 3:IF B(
Z)=E(Z) THEN CUENTA = CUENTA +1
420 IF AYUDA >0 THEN NEXT Z:IF CUENTA < AYUDA TH
EN 290
430 M=100*B(1) + 10 *B(2) + B(3)
440 K(PRUEBA) = M
450 IF PRUEBA < 3 THEN 500
460 CUENTA=1
470 CUENTA = CUENTA + 1
480 IF K(CUENTA) = M THEN 290
490 IF CUENTA < PRUEBA -1 THEN 470
500 RETURN
510 REM INICIALIZACION
520 DEFINT A - Z
530 PRUEBA =0:Q=9:AYUDA=0
540 RANDOMIZE
550 DIM B(3),C(9),E(3),K(100)
560 FOR Z= 1 TO 9
570 C(Z)=Z
580 NEXT Z
590 RETURN
600 PRINT:PRINT
610 PRINT"He acertado tu codigo: es el "B(1);B(2
);B(3)
620 PRINT TAB(5);"en solo"PRUEBA "intentos."
630 END
```

Ahora, tras alguna modificación, el ordenador trata de romper un código de cuatro números:

```
Intento numero 10          Intento numero 11

Mi intento es 9  5  1  2   Mi intento es 1  2  4  9

Cuantos negros?0           Cuantos negros?0

 cuantos blancos?1          cuantos blancos?2

      Intento numero 15

      Mi intento es 6  3  9  4

      Cuantos negros?4


      He acertado tu codigo: es el  6  3  9  4
           en solo 15 intentos.
```

Este es el listado para adivinar códigos de cuatro números, basado en el de tres números. Observe que en la línea 420 se le añade "— 1", así como otros cambios:

```
  10 REM Cerebro electronico - 4 digitos
  20 GOSUB 510:REM Inicializacion
  30 REM Intenta adivinar
  40 IF PRUEBA =0 THEN FOR Z=1 TO 4:B(Z) = Z:NEXT
Z:GOTO70
```

```
   50 IF PRUEBA=1 THEN FOR Z=1 TO 4:B(Z)=Z+4:NEXT
Z:GOTO70
   60 GOSUB 280
   70 CLS
   80 PRUEBA = PRUEBA + 1
   90 PRINT:PRINT
  100 PRINT"Intento numero" PRUEBA
  110 PRINT
  120 PRINT"Mi intento es"B(1);B(2);B(3);B(4)
  130 PRINT:PRINT
  140 INPUT"Cuantos negros";N
  150 IF N=4 THEN GOTO 600
  160 IF N=3 THEN 190
  170 PRINT:PRINT
  180 INPUT" cuantos blancos";B
  190 IF B+N=4 THEN Q=4:C(1)=B(1):C(2)=B(2):C(3)=B
(3):C(4)=B(4)
  200 IF N+B=0 THEN C(B(1))=0:C(B(2))=0:C(B(3))=0:
C(B(4))=0:GOTO30
  210 IF N>AYUDA THEN FOR Z=1 TO 4:E(Z)=B(Z):NEXTZ
:AYUDA=N
  220 FOR Z=1 TO 9
  230 FOR D=1 TO 4
  240 IF B(D) = C(Z) THEN C(Z) = C(Z) + (B + N)*10
0 + B*10
  250 NEXTD
  260 NEXT Z
  270 GOTO30
  280 REM Coje 3 numeros
  290 FOR Z= 1TO 4
  300 D1=C(INT(RND(-1)*Q)+1)
  310 IF D1 = 0 THEN 300
  320 D2= C(INT(RND(-1)*Q)+1)
  330 IF D2=0 THEN 320
  340 IF INT(D1/10) > INT(D2/10) THEN B(Z) = D1
  350 IF INT(D1/10) < INT(D2/10) THEN B(Z) = D2
  360 IF INT(D1/10) = INT(D2/10) THEN B(Z) = D1
  370 IF B(Z)>100 THEN B(Z)= B(Z)-100*INT(B(Z)/100
):GOTO370
  380 IF B(Z)>10 THEN B(Z)=B(Z)-10*INT(B(Z)/10):GO
TO 380
  390 NEXT Z
  400 IF B(1)=B(2) OR B(1)=B(3) OR B(1)=B(4) OR B(
```

```
2)=B(3)  OR B(2)=B(4) OR B(3)=B(4) THEN 290
  410 IF AYUDA >0 THEN CUENTA=0:FOR Z=1 TO 4:IF B(
Z)=E(Z) THEN CUENTA = CUENTA +1
  420 IF AYUDA >0 THEN NEXT Z:IF CUENTA < AYUDA -1
 THEN 290
  430 M=1000*B(1) + 100 *B(2) + 10*B(3)+B(4)
  440 K(PRUEBA) = M
  450 IF PRUEBA < 3 THEN 500
  460 CUENTA=1
  470 CUENTA = CUENTA + 1
  480 IF K(CUENTA) = M THEN 290
  490 IF CUENTA < PRUEBA -1 THEN 470
  500 RETURN
  510 REM INICIALIZACION
  520 DEFINT A - Z
  530 PRUEBA =0:Q=9:AYUDA=0
  540 RANDOMIZE
  550 DIM B(4),C(9),E(4),K(100)
  560 FOR Z= 1 TO 9
  570 C(Z)=Z
  580 NEXT Z
  590 RETURN
  600 PRINT:PRINT
  610 PRINT"He acertado tu codigo: es el "B(1);B(2
);B(3);B(4)
  620 PRINT TAB(5);"en solo"PRUEBA"intentos."
  630 END
```

PROBOSCIDEOS

Es éste un programa en el que el ordenador muestra la buena memoria que tiene retando al jugador humano a jugar una partida de «Concentración».

En primer lugar se baraja el mazo de cartas y luego se extienden éstas boca abajo en la pantalla. El ordenador y el jugador humano escogen dos por turno. Estas se vuelven hacia arriba y si forman una pareja se le da un punto al jugador que las ha escogido. Este proceso continúa hasta que se hayan escogido todas las cartas. El ganador es el que haya seleccionado más parejas.

Lo que viene a continuación es una partida en la que (como de costumbre) el jugador humano (yo mismo) es totalmente derrotado por la memoria de 64K del ordenador:

```
Un momento, por favor,estoy barajando!

empieza a ejercitar los musculos de la memoria...

Nivel de dificultad (1 - 10)
(10 es facil, 1 casi IMPOSIBLE!)
?1
```

```
Mi puntuacion es 0 y
   la tuya es 0

Estamos jugando en el nivel 1

   1  --  2  --  3  --  4  --  5  --
   6  --  7  --  8  --  9  --  10  --
 11  --  12  --  13  --  14  --  15  --
 16  --  17  --  18  --  19  --  20  --
 21  --  22  --  23  --  24  --  25  --
 26  --  27  8D  28  --  29  --  30  --
 31  --  32  --  33  --  34  --  35  --
 36  --  37  --  38  --  39  --  40  --
 41  --  42  --  43  JC  44  --  45  --
 46  --  47  --  48  --  49  --  50  --
 51  --  52  --

Tus cartas: 27 y 43
No, no son una pareja
```

```
Mi puntuacion es 0 y
   la tuya es 0

Estamos jugando en el nivel 1

   1  --  2  --  3  --  4  --  5  --
   6  --  7  --  8  --  9  --  10  --
 11  --  12  --  13  --  14  --  15  --
 16  --  17  --  18  --  19  2C  20  --
 21  --  22  --  23  --  24  --  25  --
 26  --  27  --  28  --  29  --  30  --
 31  --  32  --  33  --  34  --  35  --
 36  --  37  --  38  --  39  --  40  --
 41  --  42  --  43  --  44  AD  45  --
 46  --  47  --  48  --  49  --  50  --
 51  --  52  --

Mis cartas: 44 y 19
Pero no son una pareja!
```

```
Mi puntuacion es 0 y
    la tuya es 0

Estamos jugando en el nivel 1

   1  --  2  --  3  --  4  --  5  --
   6  --  7  --  8  --  9  --  10  AD
 11  --  12  --  13  --  14  --  15  --
 16  --  17  --  18  --  19  --  20  6T
 21  --  22  --  23  --  24  --  25  --
 26  --  27  --  28  --  29  --  30  --
 31  --  32  --  33  --  34  --  35  --
 36  --  37  --  38  --  39  --  40  --
 41  --  42  --  43  --  44  --  45  --
 46  --  47  --  48  --  49  --  50  --
 51  --  52  --

Tus cartas: 10 y 20
No, no son una pareja

   1  8T  2  9P  3  JD  4  8P  5  AP
   6  2D  7  KC  8  DT  9  6D  10  4C
 11  3P  12  DD  13  4P  14  JC  15  AT
 16  7T  17  AD  18  3D  19  4T  20  7P
 21  6T  22  KT  23  5T  24  3T  25  QC
 26  4D  27  5D  28  8C  29  3C  30  6C
 31  KP  32  8D  33  JP  34  2P  35  9T
 36  DP  37  7C  38  2T  39  KD  40  QD
 41  9C  42  9D  43  QT  44  QP  45  6P
 46  2C  47  7D  48  5P  49  5C  50  JT
 51  AC  52  DC

Tus cartas: 30 y 45

Y asi llegamos al final
de otra partida de Proboscideo

Yo soy el ganador, humano!
```

Es posible que el jugador quiera hacer el juego más fácil, al menos al principio, añadiéndole un mecanismo que haga desaparecer de la pantalla las cartas que ya se han adivinado. Esto puede hacerse cambiando el tercer elemento de las series que representan una carta correctamente escogida. Y este es el listado de PROBOSCIDEOS:

```
   10 REM Proboscideos
   20 GOSUB 790: REM Inicializa/baraja
   30 GOSUB 610:REM Impresion
   40 GOSUB 430:REM Intento del humano
   50 GOSUB 150:REM Intento de la computadora
   60 IF HS + CS <26 THEN 30
   70 PRINT"Y asi llegamos al final"
   80 PRINT"de otra partida de Proboscideo"
   90 PRINT
  100 IF HS > CS THEN PRINT "Tu eres el vencedor!"
  110 IF CS > HS THEN PRINT"Yo soy el ganador, hum
ano!"
  120 IF CS = HS THEN PRINT"Es empate, humanoide!"
  130 END
  140 REM ****************************************
****************************
  150 REM Intento de la computadora
  160 MOVIMIENTO = 0
  170 IF R=1 OR RND(-1)<NIVEL THEN 260
  180 X=R
  190 X=X-1
  200 Y=1
  210 Y=Y+1
  220 IF D(X)=D(Y) THEN 240
  230 IF ASC(A$(D(X))) = ASC(A$(D(Y))) THEN IF X <
> Y THEN IF MID$(A$(D(X)),3,1)="X" AND MID$(A$(D(Y
)),3,1) = "X" THEN A=D(X):B=D(Y):GOTO340
  240 IF Y<X THEN 210
  250 IF X-1>R THEN 190
  260 REM Intento al azar
  270 CUENTA=0
  280 CUENTA=CUENTA+1
  290 IF CUENTA = 200 THEN 70
  300 A=INT(RND(-1)*52)+1
  310 B=INT(RND(-1)*52)+1
```

```
320 IF B=A THEN 310
330 IF RIGHT$(A$(A),1) <> "X" OR RIGHT$(A$(B),1)
 <> "X" THEN 280
340 PRINT"Yo elijo las cartas"A" y"B
350 FOR P= 1 TO 1000 :NEXT P
360 MID$(A$(A),3,1) = " ":MID$(A$(B),3,1) = " "
370 GOSUB 610
380 IF LEFT$(A$(A),1) = LEFT$(A$(B),1) THEN PRIN
T"Y son pareja!":CS=CS+1
390 IF LEFT$(A$(A),1)<> LEFT$(A$(B),1) THEN PRIN
T"Pero no son una pareja!":MID$(A$(A),3,1) = "X":M
ID$(A$(B),3,1) = "X":D$(R)=A$(A):D$(R+1)=A$(B):D(R
)=A:D(R+1)=B:R=R+2
400 FOR P=1 TO 1000:NEXT P
410 RETURN
420 REM ****************************************
***************************
430 REM Intento del humano
440 MOVIMIENTO = 1
450 PRINT:INPUT"Dime la primera carta que has el
egido";A
460 IF A<1 OR A>52 THEN 450
470 IF RIGHT$(A$(A),1) = " " THEN PRINT"Esa cart
a ya ha sido cogida":GOTO450
480 MID$(A$(A),3,1) = " "
490 INPUT"Dime la segunda";B
500 IF B<1 OR B> 52 OR B=A THEN 490
510 IF RIGHT$(A$(B),1) = " " THEN PRINT"Esa cart
a ha sido cogida":GOTO490
520 MID$(A$(B),3,1) = " "
530 GOSUB 610
540 IF LEFT$(A$(A),1) = LEFT$(A$(B),1) THEN PRIN
T "Si, son una pareja!":HS = HS + 1
550 IF LEFT$(A$(A),1) <> LEFT$(A$(B),1) THEN PRI
NT"No, no son una pareja":MID$(A$(A),3,1)= "X":MID
$(A$(B),3,1)= "X"
560 IF LEFT$(A$(A),1)<>LEFT$(A$(B),1) THEN D$(R)
=A$(A):D$(R+1)=A$(B):D(R)=A:D(R+1)=B:R=R+2
570 FOR P=1 TO 1000:NEXT P
580 GOSUB 610
590 RETURN
600 REM ****************************************
***************************
```

```
610 REM Impresion
620 CLS:PRINT"Mi puntuacion es"CS"y":PRINT"    1
a tuya es"HS
630 PRINT:PRINT"Estamos jugando en el nivel"NIVE
L*10
640 PRINT:PRINT"  ";
650 FOR Z=1 TO 52
660 IF Z=6 THEN PRINT"  ";
670 PRINTZ;" ";
680 B$=RIGHT$(A$(Z),1)
690 IF B$="X" THEN PRINT"-- ";
700 IF B$=" " THEN PRINT LEFT$(A$(Z),2);" ";
710 IF 5*(INT(Z/5)) = Z THEN PRINT
720 NEXT Z
730 PRINT
740 PRINT
750 IF MOVIMIENTO = 1 THEN PRINT"Tus cartas:"A"y
"B
760 IF MOVIMIENTO =0 THEN PRINT"Mis cartas:"A"y"
B
770 RETURN
780 REM ****************************************
****************************
790 REM Inicializa/baraja
800 RANDOMIZE
810 DIM A$(52),D$(500),D(500)
820 CLS:PRINT"Un momento, por favor,estoy baraja
ndo!"
830 PRINT:PRINT"empieza a ejercitar los musculos
 de la memoria..."
840 CS=0:HS=0:R=1:MOVIMIENTO=3
850 FOR B=1 TO 52
860 D$(B)=","
870 READA$(B)
880 NEXT B
890 REM Rutina para barajar de Moses/Oakford
900 FOR J= 52 TO 2 STEP-1
910 Z=INT(RND(-1)*J)+1
920 H$=A$(Z)
930 A$(Z)=A$(J)
940 A$(J)=H$
950 NEXT J
960 PRINT:PRINT"Nivel de dificultad (1 - 10)"
```

```
  970 PRINT"(10 es facil, 1 casi IMPOSIBLE!)"
  980 INPUT NIVEL:IF NIVEL < 1 OR NIVEL > 10 THEN
980
  990 NIVEL = NIVEL/10
 1000 RETURN
 1010 DATA"ATX","2TX","3TX","4TX","5TX","6TX","7TX
","8TX","9TX","DTX","JTX","QTX","KTX"
 1020 DATA"ADX","2DX","3DX","4DX","5DX","6DX","7DX
","8DX","9DX","DDX","JDX","QDX","KDX"
 1030 DATA"APX","2PX","3PX","4PX","5PX","6PX","7PX
","8PX","9PX","DPX","JPX","QPX","KPX"
 1040 DATA"ACX","2CX","3CX","4CX","5CX","6CX","7CX
","8CX","9CX","DCX","JCX","QCX","KCX"
```

AMANUENSE

A continuación tenemos nuestro programa poético. En este no es necesaria la acción del jugador o usuario, salvo para admirar el maravilloso resultado. Este es un ejemplo del tipo de poemas que puede producir:

```
EL ALCAIDE IMPLORO SIN LIMITE
      EN EL TERRENO ENEMIGO...
MIRANDO FIJAMENTE RENUNCIAR,
  PARA DESVIAR EL HADA...
 VACIANDO, LUEGO HIRIENDO
    ...ENCERRANDO, CARCOMIENDO.

EL APLICANTE URGIO HUMILDEMENTE
      EN AQUEL BABEL...
CREYENDO EN VACILAR,
  PARA VACILAR EL ESPECTRO...
 VACIANDO, LUEGO ROMPIENDO
    ...DESAFIANDO, DANANDO.

EL CENTINELA EXIGIO ENCORAGINADO
      EN EL ESCENARIO DE LA BATALLA...
LLORANDO POR OBJETAR,
  PARA DESVIAR EL FAUNO...
 DERROCHANDO, LUEGO MINANDO
    ...AMORDAZANDO, MARCHITANDO.
```

EL MENDICANTE CONJURO TONTAMENTE
EN EL CAMPO DE BATALLA...
OBSERVANDO ABJURAR,
PARA OBJETAR EL ESPECTRO...
MALGASTANDO, LUEGO EROSIONANDO
...ENCERRANDO, PUDRIENDO.

EL MENDICANTE CONJURO DESASISTIDO
EN EL TERRENO DE JUEGO...
ESPERANDO DESISTIR,
PARA SOFOCAR EL DUENDE...
AGOTANDO, LUEGO EROSIONANDO
...COERCIONANDO, ROMPIENDO.

EL PORDIOSERO ESPERO VACILANTEMENTE
EN EL ESCENARIO DE LA BATALLA...
DESEANDO ABJURAR,
PARA OBJETAR EL SATIRO...
TERMINANDO, LUEGO HIRIENDO
...RESTRINGIENDO, DAÑANDO.

EL APLICANTE URGIO ORGULLOSAMENTE
EN EL JUICIO...
DESEANDO RETRACTAR,
PARA ABJURAR EL SATIRO...
VACIANDO, LUEGO PUDRIENDO
...AMORDAZANDO, PUDRIENDO.

EL SUPLICANTE URGIO DESDENOSAMENTE
EN EL TERRENO DE JUEGO...
GIMIENDO POR VACILAR,
PARA OBJETAR EL PASADO...
AGOTANDO, LUEGO MINANDO
...MANIATANDO, MINANDO.

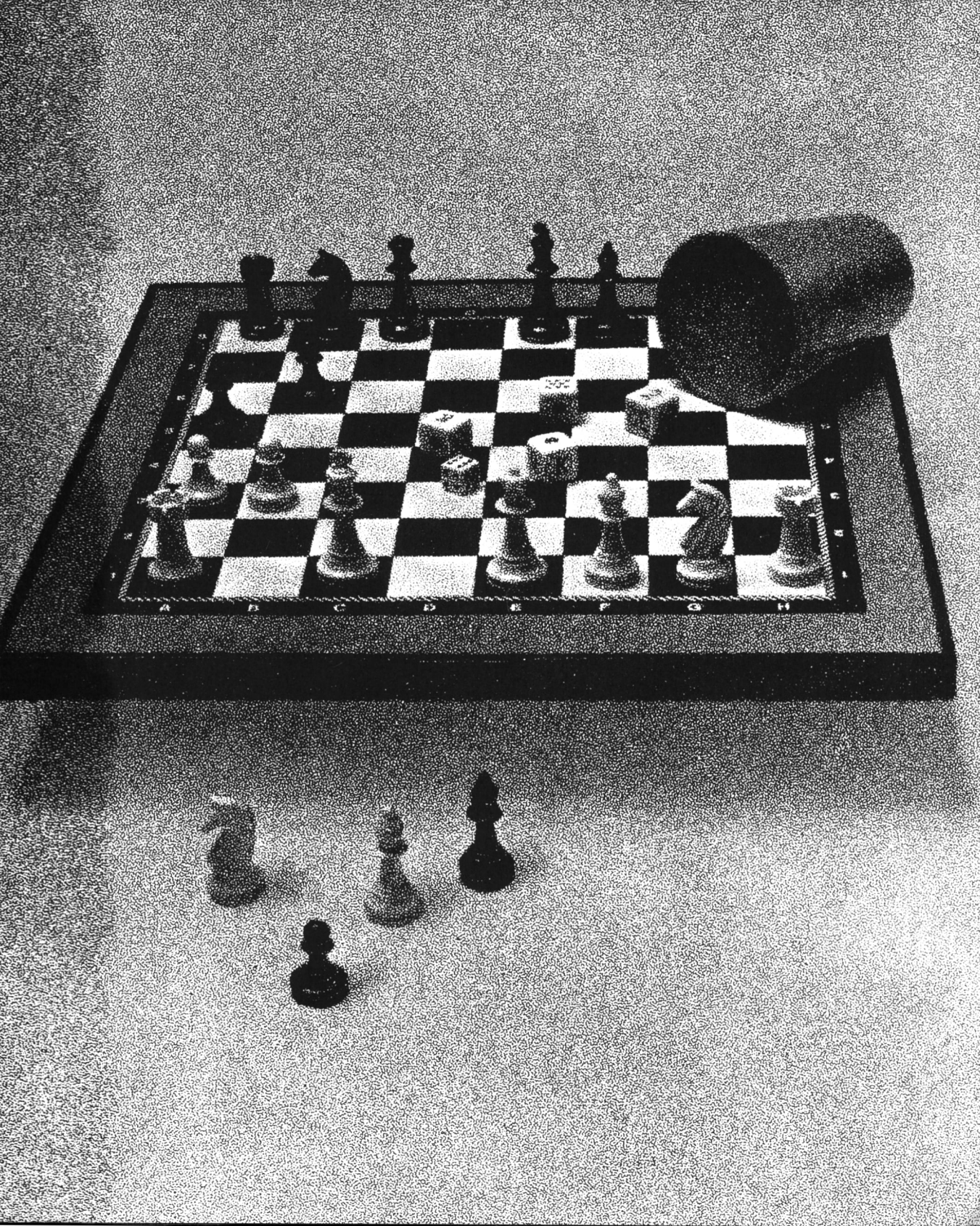
```
EL ALCAIDE ANHELO DESDENOSAMENTE
      EN EL ESCENARIO DE LA BATALLA...
DESEANDO VACILAR,
  PARA ABJURAR EL FAUNO...
 MALGASTANDO, LUEGO CORROYENDO
    ...AMORDAZANDO, DANANDO.
```

Este es el listado de AMANUENSE. Una vez que lo haya ejecutado varias veces, puede que quiera cambiar las palabras de las sentencias DATA, para modificar el tipo de poesía que produce:

```
   10 REM Amanuense
   20 CLS
   30 DIM A$ (12)
   40 N=0
   50 PRINT"Aprieta cualquier tecla para empezar"
   60 N=N+1:IF INKEY$=""THEN 60
   70 RANDOMIZE:CLS
   80 REM ****************************************
***********************
   90 RESTORE 330:GOSUB 450:A$(1)=B$
  100 RESTORE 340:GOSUB 450:A$(2)=B$
  110 RESTORE 350:GOSUB 450:A$(3)=B$
  120 RESTORE 360:GOSUB 450:A$(4)=B$
  130 RESTORE 370:GOSUB 450:A$(5)=B$
  140 RESTORE 380:GOSUB 450:A$(6)=B$
  150 RESTORE 390:GOSUB 450:A$(7)=B$
  160 RESTORE 400:GOSUB 450:A$(8)=B$
  170 RESTORE 410:GOSUB 450:A$(9)=B$
  180 RESTORE 420:GOSUB 450:A$(10)=B$
  190 RESTORE 430:GOSUB 450:A$(11)=B$
  200 RESTORE 420:GOSUB 450:A$(12)=B$
  210 REM ****************************************
***********************
  220 PRINT"EL ";A$(1);" ";A$(2);" ";A$(3)
  230 PRINT"       EN ";A$(4);"..."
  240 PRINTA$(5);" ";A$(6);","
  250 PRINT"  PARA ";A$(7);" EL ";A$(8);"..."
```

```
  260 PRINT" ";A$(9);", LUEGO ";A$(10)
  270 PRINT"    ...";A$(11);", ";A$(12);"."
  280 PRINT:PRINT
  290 REM ****************************************
************************
  300 FOR T=1 TO 1000:NEXT T
  310 GOTO90
  320 REM ****************************************
************************
  330 DATA"APLICANTE","SUPLICANTE","MENDICANTE","P
ORDIOSERO","ASPIRANTE","RECLAMANTE","POSTULANTE","
DIRIGENTE","CENTINELA","ALCAIDE"
  340 DATA"IMPLORO","SUPLICO","ESPERO","ROGO","CON
JURO","PRESIONO","URGIO","ANHELO","DEMANDO","EXIGI
O"
  350 DATA"CON EXITO","ENCORAGINADO","PROGRESIVAME
NTE","TONTAMENTE","SIN LIMITE","DESASISTIDO","VACI
LANTEMENTE","HUMILDEMENTE","DESDENOSAMENTE","ORGUL
LOSAMENTE"
  360 DATA"EL CAMPO","EL CAMPO DE BATALLA","EL TER
RENO DE JUEGO","LA PISTA DEL CIRCO","AQUEL BABEL",
"EL TERRENO ENEMIGO","EL HIPODROMO","EL ESCENARIO
DE LA BATALLA","EL JUICIO","LA PISTA DE CARRERAS"
  370 DATA"ESPERANDO","PREGUNTANDO A","CREYENDO EN
","OBSERVANDO","ORANDO POR","PERMANECIENDO POR","L
LORANDO POR","DESEANDO","MIRANDO FIJAMENTE","GIMIE
NDO POR"
  380 "LA FATALIDAD","LA RUEDA DE LA FORTUNA","LA
GRAN DECISION","LAS ESTRELLAS","UNA INFLUENCIA AST
RAL","UNA OPORTUNIDAD IRREPETIBLE","UN HECHICERO",
"UN ESPIRITU ATORMENTADO","UN FANATICO","UN BRUJO"
  390 DATA"SOFOCAR","RETRACTAR","ABJURAR","DESDECI
R","RENUNCIAR","DESISTIR","VACILAR","OBJETAR","DES
VIAR","RETROCEDER"
  400 DATA"FANTASMA","ESPECTRO","DUENDE","FAUNO","
SATIRO","HADA","DRIADE","TRASGO","COMPLEJO","PASAD
O"
  410 DATA"MALGASTANDO","GASTANDO","USANDO","CONSU
MIENDO","AGOTANDO","DESECANDO","VACIANDO","TERMINA
NDO","DERROCHANDO","DISIPANDO"
  420 DATA"CORROYENDO","EROSIONANDO","MARCHITANDO"
,"PUDRIENDO","CARCOMIENDO","MINANDO","AGITANDO","R
OMPIENDO","DANANDO","HIRIENDO"
```

```
430 DATA"RESTRINGIENDO","COERCIONANDO","OBSTACUL
IZANDO","RETRASANDO","ENCERRANDO","IMPONIENDO","MA
NIATANDO","AMORDAZANDO","DESAFIANDO","SUPRIMIENDO"
440 REM ****************************************
**************************
450 FOR N = 1 TO (RND(-1)*10+1)
460 READ B$
470 NEXT N
480 RETURN
```


A B C D E F G H

SOLO PARA DIVERTIRSE

Hay una serie de juegos, como los incluidos en este capítulo del libro, que están sólo pensados y diseñados con el fin de divertir al jugador. Estoy seguro que el lector se lo pasará en grande jugando a CEROS Y CRUCES o RESORTE OCULTO pasando por EL CAMPO MINADO.

Además de los juegos mencionados más arriba, se incluyen en este capítulo NIMGRAB y CADDY. Y si quiere divertir a un grupo de chavales dejándoles jugar con el ordenador durante un rato, tenemos lo necesario: un programa inmenso llamado CABALGATA DE PUZZLES compuesto por toda una colección de divertidísimos juegos. En éste sólo se puede pasar al siguiente juego cuando uno ha terminado el anterior habiendo ganado. Es un medio ideal de mantener entretenidos a los más jóvenes (en edad o espíritu).

Y sin más preámbulos, pasemos a ellos de inmediato.

RESORTE OCULTO

Empezaremos esta sección del libro con uno de los juegos más simples de los que componen este grupo, RESORTE OCULTO. Como explica el programa, el ordenador produce dos números entre el 1 y el 13 y le pide al jugador que apueste a ver si adivina el siguiente número que piensa poner entre los dos primeros. Es muy fácil de jugar y muy divertido.

Esto sería una partida en acción:

```
Mi primer numero es 2
     El segundo es 6

Tienes $ 10

Cuanto quieres apostar
a que mi proximo numero
esta entre 2  y 6 ?5

El numero ha sido 4
Bien hecho, ganas $ 10
```

Y este es el listado:

```
 10 REM Resorte oculto
 20 GOSUB 330
 30 GOSUB 80
 40 IF D<1 THEN GOTO 420
 50 GOSUB 340
 60 GOTO30
 70 REM ****************************************
************************
 80 PRINT:PRINT:PRINT
 90 PRINT"Mi primer numero es"A
100 PRINT"       El segundo es"B
110 PRINT
120 PRINT"Tienes $"D
130 PRINT
140 PRINT"Cuanto quieres apostar"
150 PRINT"a que mi proximo numero"
160 PRINT"esta entre"A" y"B;
170 INPUT E
180 IF E>D THEN 170
190 D=D-E
200 GOSUB 500
210 PRINT:PRINT"El numero ha sido"C
220 GOSUB 500
230 IF C<A AND C<B OR C>A AND C>B THEN 290
240 PRINT"Bien hecho, ganas $"2*E
250 D=D+3*E
260 GOSUB 500
270 RETURN
280 REM ****************************************
************************
290 PRINT"Lo siento, pierdes $"E
300 GOSUB 500
310 RETURN
320 REM ****************************************
************************
330 D=20
340 CLS
350 A= INT(RND(-1)*13)+1
360 B=INT(RND(-1)*13)+1
370 IF ABS(A-B)<2 OR ABS(A-B)>6 THEN 360
```

```
380 C=INT(RND(-1)*13)+1
390 IF A=C OR B=C THEN 380
400 RETURN
410 REM ****************************************
*************************
420 PRINT
430 PRINT"EL JUEGO HA TERMINADO"
450 PRINT"  ESTAS ARRUINADO!"
460 PRINT
470 PRINT"GRACIAS POR EL JUEGO"
480 END
490 REM ****************************************
*************************
500 FOR Z=1 TO 1000
510 NEXT Z
520 RETURN
```

NIMGRAB

NIMGRAB es un juego que parece muy sencillo a primera vista, pero que es endemoniado a la hora de jugar y mucho más todavía si el jugador pretende ganar. El jugador humano y el taimado ordenador van retirando por turno los objetos que aparecen en la pantalla. Hay un límite que marca cuántos objetos se pueden retirar en cada jugada (y este límite cambia de una partida a otra, aunque no cambia dentro de la misma partida). El ganador es aquel que fuerza a su contrincante a coger el último objeto que queda en la pantalla.

En cuanto haya visto el programa en acción, comprenderá cómo se juega sin más explicaciones. Así que ya puede empezar a trabajar perfeccionando el método de vencer al ordenador. Lo que sigue es una partida de NIMGRAB:

```
El mazimo que puedes quitar son 4

*****************************************************

 1  2  3  4  5  6  7
 8  9  10  11
 12  13
 14  15  16  17
*****************************************************

Cuantos quieres coger?

Asi que quieres coger 2

*****************************************************
```

```
*****************************************************

El maximo que puedes quitar son 4

*****************************************************

Has cogido 2 , y yo cogi 4
 1  2  3
 4  5  6  7  8  9  10  11

*****************************************************

Cuantos quieres coger?

Asi que quieres coger 3

*****************************************************

*****************************************************

El maximo que puedes quitar son 4

*****************************************************

Has cogido 4 , y yo cogi 1
 1

*****************************************************

Cuantos quieres coger?

Asi que quieres coger 1

*****************************************************

Has cogido el ultimo
              asi que gano yo!
```

Y este es el listado:

```
 10 REM Nimgrab
 20 CLS
 30 PRINT"Oprime 'E' para empezar el juego..."
 40 N=0
 50 N=N+1:IF INKEY$="" THEN 50
 60 RANDOMIZE
 70 CLS
 80 M=0:E=0:Z=INT(RND(-N)*8)+16
 90 IF 2*INT(Z/2)=Z THEN Z=Z+1
100 H=3 + INT(RND(-N)*2)
110 PRINT"El maximo que puedes quitar son"H
120 GOSUB 320
130 IF E>0 THEN PRINT"Has cogido"E", y yo cogi"Q
140 FOR K=1 TO Z
150 PRINTK;:IF RND(-N)>.8 THEN PRINT
160 NEXT K
170 GOSUB 320
180 PRINT"Cuantos quieres coger?"
190 E=VAL(INKEY$)
200 IF E<1 OR E>H OR E>Z THEN 190
210 PRINT:PRINT"Asi que quieres coger"E
220 Z=Z-E
230 GOSUB 320
240 IF Z<1 THEN PRINT"Has cogido el ultimo":PRIN
T,"asi que gano yo!":END
250 Q = Z - 1 - INT((Z-1)/(H+1))*(H+1) - INT(RND
(-N)*2) + INT(RND(-N)*2)
260 IF Q<1 OR Q>H THEN 250
270 GOSUB 320
280 Z=Z-Q
290 IF Z<1 THEN PRINT"Yo he cogido el ultimo":PR
INT,"asi que tu ganas!!":END
300 GOSUB 320
310 GOTO110
320 PRINT
330 PRINT"*****************************************#
*************************"
340 PRINT
350 FOR O=1 TO 200:NEXT O
360 RETURN
```

CADDY

Creo que ha llegado el momento de que usted, lector, y su versátil ordenador, jueguen un rato al golf, lo que podrán hacer con este programa, CADDY.

El jugador humano y el ordenador han de vérselas con nueve hoyos y, como verá cuando ejecute el programa, el ordenador se muestra de lo más educado llevando por usted la cuenta de sus puntos. Después de cada hoyo le informará de cómo va hasta ese momento y calculará la media de sus marcas por hoyo. Todo lo que el jugador tiene que hacer es golpear la bola. Si el jugador golpea la bola demasiado lejos, el ordenador se asegurará automáticamente de que el siguiente golpe retrocede hacia el hoyo.

Se dará cuenta en seguida de que el juego requiere cierta habilidad especialmente en los hoyos que presentan un alto factor de dificultad.

Aquí tiene una partida de CADDY:

```
<<<< Hoyo numero 1 >>>>

EL FACTOR DE DIFICULTAD ES TRES
                    O
##########################\ /#########
########################### ##########
^^^
```

```
<<<< Hoyo numero 1 >>>>

EL FACTOR DE DIFICULTAD ES CUATRO
                                  o
##########################\ /#########
########################## #########
^^^
```

```
Final de la vuelta, campeon!

Tu puntuacion ha sido 35
Y la media por hoyo fue 3

Aprieta la 'S' para otra vuelta, o 'N' para acabar.
```

Y este es el listado:

```
 10 REM Caddy
 20 DIM X(9):CO = 0
 30 RANDOMIZE
 40 FOR Z=1 TO 9
 50 SC=0
 60 J=INT(RND(-1)*12)
 70 Q=INT(RND(-1)*3)+3
 80 IF Q=5 THEN Q$="CINCO"
 90 IF Q=4 THEN Q$="CUATRO"
100 IF Q=3 THEN Q$="TRES"
110 CLS:PRINT:PRINT
120 IF Z=2 THEN PRINT"La puntuacion hasta este h
oyo es"X(1)
130 IF Z>2 THEN PRINT"La puntuacion hasta este h
oyo es"K
140 PRINT"<<<< Hoyo numero"Z">>>>"
150 PRINT:PRINT"EL FACTOR DE DIFICULTAD ES "Q$
160 GOSUB 430
170 PRINT:INPUT"Dime la fuerza del golpe";A
180 IF J> 26 THEN A=-A
190 J=J+INT(A/(RND(-1)*Q+1))
200 IF J = 26 THEN GOSUB 490
210 IF J<> 26 THEN GOSUB 430
220 SC=SC+1
230 PRINT"Despues de este golpe, tu puntuacion e
s"SC
240 FOR P=1 TO 500:NEXT P
250 IF J<>26 THEN 110
```

```
260 C=C+SC
270 X(Z)=SC
280 IF Z=1 THEN 390
290 K=0
300 PRINT"El juego hasta ahora:"
310 FOR J=1 TO Z
320 K=K+X(J)
330 PRINT"Hoyo"J"necesito"X(J)"golpes"
340 FOR M=1 TO 300:NEXT M
350 NEXT J
360 IF Z< 9 THEN PRINT:PRINT"La media por ahora
es de"INT((K+.5)/Z)
370 FOR P=1 TO 1000:NEXT P
380 IF Z>1 THEN PRINT:PRINT"La puntuacion a los"
Z"hoyos es"C
390 IF Z=1 THEN PRINT:PRINT"La puntuacion para e
l primer hoyo es"C
400 FOR M=1 TO 1000:NEXT M
410 NEXT Z
420 GOTO560
430 IF J> 30 THEN J=30
440 PRINT TAB(J+1);"o"
450 PRINT"##########################\ /#########
"
460 PRINT"########################## ##########
"
470 PRINT"^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
"
480 RETURN
490 PRINT:PRINT:PRINT"Lo has conseguido!"
500 FOR P=1 TO 100:NEXT P
510 PRINT"##########################\ /#########
"
520 PRINT"############################o##########
"
530 PRINT"^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
"
540 FOR P=1 TO 2000:NEXT P
550 RETURN
560 PRINT:PRINT"Final de la vuelta, campeon!"
570 PRINT:PRINT"Tu puntuacion ha sido"C
580 PRINT"Y la media por hoyo fue"INT((C+.5)/9)
590 PRINT:PRINT
```

```
600 PRINT"Aprieta la 'S' para otra vuelta, o 'N'
 para acabar."
610 A$=INKEY$
620 IF A$<>"S" AND A$<>"s" AND A$<>"N" AND A$<>"
n" THEN 610
630 IF A$="S" OR A$="s" THEN 10
640 PRINT:PRINT"OK, gracias por jugar, campeon":
END
```

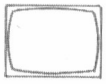
CABALGATA DE PUZZLES

Este juego no requiere ninguna explicación. Las reglas están todas contenidas en el programa que presenta, además, toda una divertida sucesión de actividades y juegos. Como descubrirá el jugador en su momento, uno no puede pasar al siguiente juego antes de completar el anterior.

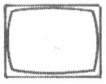
El listado es bastante largo, pero es fácil convencerse de que merece la pena si se piensa en todos los agradables momentos que se podrán disfrutar una vez que se tiene introducido. Cada vez que un amigo le diga aquello de lo maravilloso que es disponer de un ordenador personal para después añadir que el problema es saber lo que se puede hacer con él, todo lo que tendrá que hacer el lector es meter este programa y después irse a leer un libro tranquilamente mientras el ordenador se encarga de entretener y convencer al amigo en cuestión.

Lo que sigue es un ejemplo del funcionamiento de CABALGATA DE PUZZLES:

```
Piensa un numero entre
uno y sesenta y tres
```



```
Ahora te voy a mostrar cinco
pantallas de numeros.
```



Si tu numero aparece en la pantalla,
limitate a apretar la 'S'.
Si no esta en la pantalla,
limitate a apretar la 'N'

Voy a decirte el numero que
habias pensado, al final del
juego...

```
1 3 5 7 9 11 13 15 17
19 21 23 25 27 29 31
33 35 37 39 41 43 45
47 49 51 53 55 57 59
61 63

Oprime 'S' si el numero
esta aqui, 'N' si no es asi
```

```
 2 3 6 7 10 11 14 15
18 19 22 23 26 27 30
31 34 35 38 39 42 43
46 47 50 51 54 55 58
59 62 63

Oprime 'S' si el numero
esta aqui, 'N' si no es asi
```

```
 4 5 6 7 12 13 14 15
20 21 22 23 28 29 30
31 36 37 38 39 44 45
46 47 52 53 54 55 60
61 62 63

Oprime 'S' si el numero
esta aqui, 'N' si no es asi
```

```
8 9 10 11 12 13 14 15
24 25 26 27 28 29 30
31 40 41 42 43 44 45
46 47 56 57 58 59 60
61 62 63

Oprime 'S' si el numero
esta aqui, 'N' si no es asi
```

```
16 17 18 19 20 21 22
23 24 25 26 27 28 29
30 31 48 49 50 51 52
53 54 55 56 57 58 59
60 61 62 63
```

```
32 33 34 35 36 37 38
39 40 41 42 43 44 45
46 47 48 49 50 51 52
53 54 55 56 57 58 59
60 61 62 63
```

```
Oprime 'S' si el numero
esta aqui, 'N' si no es asi
```

```
Oprime 'S' si el numero
esta aqui, 'N' si no es asi
```

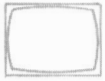
```
Tu numero era...

1 !
```

```
Voy a pensar un numero
entre uno y 40 y tu
tienes solo seis oportunidades
de acertarlo...si lo consiques,
pasas al siguiente juego
```

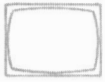
```
Si no lo consigues,
te dare una oportunidad mas facil
hasta que lo consigas... y asi
sucesivamente,hasta que incluso
alguien tan estupido como un humano
pueda conseguirlo...
```

```
Espera un momento...
```

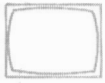
OK, estoy pensando mi numero
entre uno y 40



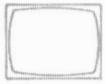
Adelante con el intento No 1
?10
Demasiado bajo!



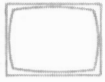
Adelante con el intento No 2
?30
Demasiado alto!



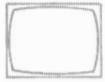
Adelante con el intento No 3
?27

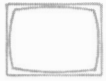


Yupiii!!! vivaa!!!

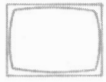


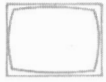
Lo has conseguido, muchacho...





Es el momento de
asombrarte...





Sigue estas instrucciones,
y oprime RETURN despues
de haberlas ejecutado...

Empieza por multiplicar tu
edad por dos...
?

Ahora sumale cinco...
?

Ahora multiplicalo por 50...
?

Ahora restale 365...
?

Ahora sumale la cantidad de dinero
suelto que lleves en el bolsillo...
?

Ahora dame el numero
que ha resultado...
?1905

Tienes 20 pesetas sueltas...

y tienes 20 anos...

Pongamos tu cerebro a
trabajar de nuevo...

En este juego, empezamos
con dos juegos de cuatro
objetos en linea, con
un espacio entre ellos,
algo asi:

$$$$ ****

$$$*$* **

Y asi sucesivamente....

Aprieta RETURN cuando
estes preparado...
?

Tienes que conseguir que
aparezcan asi:
**** $$$$
en el tiempo mas corto
posible...

Solo puedes moverte en un
espacio vacio, o saltar sobre
otra pieza en un espacio
vacio...tal y como voy a
demostrarte...

```
$ $ $ $ $ * * * *
1 2 3 4 5 6 7 8 9

Tienes  0 correctas
Numero de la pieza que quieres mover?
```

Lo cual nos lleva al final
de nuestra cabalgata de puzzles

Aprieta 'S' para jugar de nuevo
o 'N' para acabar...

Gracias por pasar tanto
tiempo jugando conmigo...
Te aseguro que fue divertido!

Espero volver a verte...

Y este es el largo listado que había prometido:

```
 10 REM Cabalgata de puzzles
 20 GOSUB 2530
 30 X=.5:N=0
 40 PRINT"Piensa un numero entre"
 50 PRINT"uno y sesenta y tres"
 60 GOSUB 2550
 70 PRINT"Ahora te voy a mostrar cinco"
 80 PRINT"pantallas de numeros."
 90 GOSUB 2550
100 PRINT"Si tu numero aparece en la pantalla,'
110 PRINT"limitate a apretar la 'S'."
120 PRINT"Si no esta en la pantalla,"
130 PRINT"limitate a apretar la 'N'"
140 GOSUB 2550
150 PRINT"Voy a decirte el numero que "
160 PRINT"habias pensado, al final del"
170 PRINT"juego..."
180 GOSUB 2560:GOSUB 2560
190 GOSUB 2530
200 PRINT"1 3 5 7 9 11 13 15 17"
210 PRINT"19 21 23 25 27 29 31"
220 PRINT"33 35 37 39 41 43 45"
230 PRINT"47 49 51 53 55 57 59"
240 PRINT"61 63"
250 GOSUB 630
260 PRINT" 2 3 6 7 10 11 14 15"
270 PRINT"18 19 22 23 26 27 30"
280 PRINT"31 34 35 38 39 42 43"
290 PRINT"46 47 50 51 54 55 58"
300 PRINT"59 62 63"
310 GOSUB 630
320 PRINT" 4 5 6 7 12 13 14 15"
330 PRINT"20 21 22 23 28 29 30"
340 PRINT"31 36 37 38 39 44 45"
350 PRINT"46 47 52 53 54 55 60"
360 PRINT"61 62 63"
370 GOSUB 630
380 PRINT"8 9 10 11 12 13 14 15"
390 PRINT"24 25 26 27 28 29 30"
```

```
400 PRINT"31 40 41 42 43 44 45"
410 PRINT"46 47 56 57 58 59 60"
420 PRINT"61 62 63"
430 GOSUB 630
440 PRINT"16 17 18 19 20 21 22"
450 PRINT"23 24 25 26 27 28 29"
460 PRINT"30 31 48 49 50 51 52"
470 PRINT"53 54 55 56 57 58 59"
480 PRINT"60 61 62 63"
490 GOSUB 630
500 PRINT"32 33 34 35 36 37 38"
510 PRINT"39 40 41 42 43 44 45"
520 PRINT"46 47 48 49 50 51 52"
530 PRINT"53 54 55 56 57 58 59"
540 PRINT"60 61 62 63"
550 GOSUB 630
560 GOSUB 2550:GOSUB 2550
570 GOSUB 2530
580 PRINT"Tu numero era..."
590 GOSUB 2550
600 PRINT,N"!"
610 GOSUB 2550
620 GOTO730
630 X=X+X
640 GOSUB 2550
650 PRINT"Oprime 'S' si el numero"
660 PRINT"esta aqui, 'N' si no es asi"
670 A$=INKEY$
680 IF A$<>"n" AND A$<>"N" AND A$<>"S" AND A$<>"
s" THEN 670
690 IF A$="S" OR A$="s" THEN N=N+X
700 GOSUB 2530
710 RETURN
720 REM ****************************************
****************************
730 GOSUB 2530
740 PRINT"Es el momento para algo"
750 PRINT"casi totalmente diferente!"
760 GOSUB 2550
770 W=40
780 PRINT"Voy a pensar un numero"
790 PRINT"entre uno y"W"y tu"
800 PRINT"tienes solo seis oportunidades"
```

```
 810 PRINT"de acertarlo...si lo consiques,"
 820 PRINT"pasas al siguiente juego"
 830 GOSUB2550
 840 PRINT"Si no lo consigues,"
 850 PRINT"te dare una oportunidad mas facil"
 860 PRINT"hasta que lo consigas... y asi"
 870 PRINT"sucesivamente,hasta que incluso"
 880 PRINT"alguien tan estupido como un humano"
 890 PRINT"pueda conseguirlo..."
 900 PRINT:PRINT
 910 PRINT"Espera un momento..."
 920 GOSUB 2550
 930 GOSUB 2530
 940 PRINT"OK, estoy pensando mi numero"
 950 PRINT"entre uno y"W
 960 N=INT(RND(-1)*W) + 1
 970 J=0
 980 J=J+1
 990 PRINT:PRINT"Adelante con el intento No"J
1000 INPUT Q
1010 IF Q=N THEN 1140
1020 IF Q<1 OR Q>W THEN 1000
1030 IF Q<N THEN PRINT"Demasiado bajo!"
1040 IF Q>N THEN PRINT"Demasiado alto!"
1050 IF J<6 THEN 980
1060 PRINT:PRINT"No tienes remedio..."
1070 PRINT"Estaba pensando en el "N
1080 IF W=5 THEN PRINT"Eres un caso desesperado..
.vamos a probar":PRINT"algo diferente":GOSUB 2550:
GOTO 1190
1090 PRINT
1100 PRINT"Bueno, parece que tendre"
1110 PRINT"que darte otra oportunidad..."
1120 W=W/2
1130 GOTO 940
1140 GOSUB 2550
1150 PRINT"Yupiii!!!  vivaa!!!"
1160 GOSUB 2550
1170 PRINT"Lo has conseguido, muchacho..."
1180 GOSUB 2550
1190 GOSUB 2530
1200 REM ****************************************
****************************
```

```
1210 PRINT"Es el momento de"
1220 PRINT"asombrarte..."
1230 GOSUB 2550
1240 PRINT"Sigue estas instrucciones,"
1250 PRINT"y oprime RETURN despues"
1260 PRINT"de haberlas ejecutado..."
1270 GOSUB 2550
1280 GOSUB 2530
1290 PRINT"Empieza por multiplicar tu"
1300 PRINT"edad por dos..."
1310 INPUT A$
1320 GOSUB 2530
1330 PRINT"Ahora sumale cinco..."
1340 INPUT A$
1350 GOSUB 2530
1360 PRINT"Ahora multiplicalo por 50..."
1370 INPUT A$
1380 GOSUB 2530
1390 PRINT"Ahora restale 365..."
1400 INPUT A$
1410 GOSUB 2530
1420 PRINT"Ahora sumale la cantidad de dinero"
1430 PRINT"suelto que lleves en el bolsillo..."
1440 INPUT A$
1450 GOSUB 2530
1460 PRINT"Ahora dame el numero"
1470 PRINT"que ha resultado..."
1480 INPUT A
1490 A=A+115
1500 B=INT(A/100)
1510 A=A-100*B
1520 GOSUB 2530
1530 PRINT"Tienes "A"pesetas sueltas..."
1540 GOSUB 2550
1550 PRINT"y tienes "B"anos..."
1560 GOSUB 2550
1570 REM ******************************************
****************************
1580 GOSUB 2530
1590 PRINT"Pongamos tu cerebro a"
1600 PRINT"trabajar de nuevo..."
1610 GOSUB 2550
1620 PRINT"En este juego, empezamos"
```

```
1630 PRINT"con dos juegos de cuatro"
1640 PRINT"objetos en linea, con"
1650 PRINT"un espacio entre ellos,"
1660 PRINT"algo asi:"
1670 PRINT
1680 PRINT"$$$$ ****"
1690 GOSUB 2550
1700 PRINT"Tienes que conseguir que"
1710 PRINT"aparezcan asi:"
1720 PRINT"**** $$$$"
1730 PRINT"en el tiempo mas corto"
1740 PRINT"posible..."
1750 GOSUB 2550
1760 PRINT"Solo puedes moverte en un"
1770 PRINT"espacio vacio, o saltar sobre"
1780 PRINT"otra pieza en un espacio"
1790 PRINT"vacio...tal y como voy a"
1800 PRINT"demostrarte..."
1810 GOSUB 2550:GOSUB 2550
1820 FOR J=1 TO 3
1830 CLS
1840 PRINT"$$$$ ****"
1850 FOR Z= 1 TO 500:NEXT Z:CLS
1860 PRINT"$$$ $****"
1870 FOR Z=1 TO 500:NEXT Z:CLS
1880 PRINT"$$$*$ ***"
1890 FOR Z= 1TO 500:NEXT Z:CLS
1900 PRINT"$$$*$* **"
1910 NEXT J
1920 PRINT:PRINT"Y asi sucesivamente...."
1930 GOSUB 2550
1940 PRINT"Aprieta RETURN cuando"
1950 PRINT"estes preparado..."
1960 INPUT A$
1970 X=-1:M=9
1980 DIMA(M)
1990 FOR Q=1 TO M
2000 IF Q<5 THEN A(Q) = ASC("$")
2010 IF Q>5 THEN A(Q) = ASC("*")
2020 NEXT Q
2030 GOSUB 2530
2040 C=0
2050 X=X+1
```

```
2060 IF X=0 THEN 2080
2070 PRINT"Ese ha sido el movimiento No "X
2080 PRINT
2090 FOR Q=1 TO M
2100 PRINT CHR$(A(Q));" ";
2110 IF Q<5 AND A(Q) = ASC("*") THEN C=C+1
2120 IF A(Q) = 0 THEN H=Q
2130 IF Q>5 AND A(Q)=ASC("$") THEN C=C+1
2140 NEXT Q
2150 PRINT:PRINT"1 2 3 4 5 6 7 8 9"
2160 PRINT
2170 PRINT:PRINT"Tienes "C"correctas"
2180 GOSUB 2220
2190 IF C=8 THEN 2260
2200 PRINT:PRINT
2210 GOTO 2030
2220 INPUT"Numero de la pieza que quieres mover";
T
2230 K=A(T)
2240 A(T)=0:A(H)=K
2250 RETURN
2260 PRINT:PRINT"Bien hecho.....Has"
2270 PRINT"conseguido resolverlo en "X"movimiento
s!"
2280 GOSUB 2550
2290 GOSUB 2530
2300 REM ****************************************
****************************
2310 PRINT"Lo cual nos lleva al final"
2320 PRINT"de nuestra cabalgata de puzzles"
2330 GOSUB 2550
2340 PRINT"Aprieta 'S' para jugar de nuevo"
2350 PRINT"o 'N' para acabar..."
2360 A$=INKEY$
2370 IF A$<>"n" AND A$<>"N" AND A$<>"S" AND A$<>"
s" THEN 2360
2380 IF A$="S" OR A$="s" THEN 2460
2390 GOSUB 2530
2400 PRINT"Gracias por pasar tanto"
2410 PRINT"tiempo jugando conmigo..."
2420 PRINT"Te aseguro que fue divertido!"
2430 GOSUB 2550
2440 PRINT"Espero volver a verte..."
```

```
2450 END
2460 GOSUB 2550
2470 PRINT"OK, ahi vamos de nuevo..."
2480 GOSUB 2550
2490 CLEAR
2500 GOTO10
2510 END
2520 REM ****************************************
**************************
2530 REM Retardo, cls, print
2540 CLS
2550 PRINT:PRINT:PRINT
2560 FOR Z=1 TO 1000:NEXT Z
2570 RETURN
```

CAMPO MINADO

Creo que ha llegado el momento de que nos enfrentemos a los horrores, terrores y peligros de CAMPO MINADO. Imagínese que va usted paseando tranquilamente al sol cuando de repente se encuentra atrapado en una zona amurallada en la que hay varios robots maníacos que no buscan sino su destrucción.

Los robots aparecen con el símbolo del dólar, la H es el jugador humano y los asteriscos (*) son minas, mortales tanto para el jugador como para los robots. Estos, que están programados bastante torpemente, pueden detectar al jugador, pero no las minas. Así pues, la salvación del jugador está en conseguir poner una o dos minas entre él y el robot de modo que éste cuando quiera acercarse al jugador caerá en ellas y desaparecerá.

Los robots tienen un truco: se sacan otros robots de la manga. Es decir, que el jugador puede pensar que sólo le persigue un robot y en el último momento ver que éste se convierte en dos o más, tomando cada uno de ellos direcciones distintas con el fin de acorralarle. Este no es un juego indicado para quienes gusten de la vida tranquila y sin sobresaltos.

El jugador se puede mover hacia el norte, el sur, el este o el oeste (el norte está en la parte superior de la pantalla) e indica al ordenador la dirección hacia la que se quiere dirigir introduciendo la inicial de esa dirección (de modo que introduce "N" si quiere ir hacia el norte, "S" hacia el sur y así sucesivamente).

Observe esta partida y se dará cuenta de que el jugador no tiene verdaderamente muchas posibilidades de salir con vida:

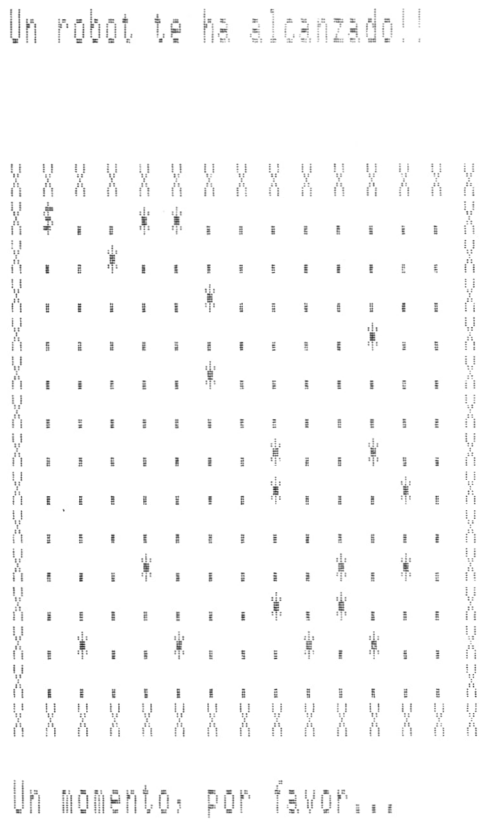
Un momento, por favor...

```
X X X X X X X X X X X X X X
X . . * . . * . . . . * . . X
X . . . . . . . . . . . . . X
X . . . . . . . . $ . . . . X
X . . * . . . . . . . . . . X
X * . . . $ . . . . . . . . X
X . . . * . . . * . . . . . X
X . . * . . . . . . . . . . X
X . . . $ . * . . . . . . . X
X . * . . . . . . * . * . . X
X . . . . . . . . . . . . . X
X . . . . . . H * . . . . . X
X . . . . . . . . * * . . . X
X . . . . . . . . $ . . . . X
X X X X X X X X X X X X X X
```

Has chocado con una mina!!

```
X X X X X X X X X X X X X X
X * . . * . . . * * . . . . X
X . . . . . . . . . . . . . X
X . . . . . . . . . . . . . X
X . . . . * * . . . . . . . X
X . . * . . . . . $ * . . . X
X . * . . . . . . . * . . . X
X * . . * . * . . . * . . . X
X . . . . . . . . . . . . . X
X * . . . . . . * . . * . . X
X * . . . . . * . . . . . . X
X . . . . . $ . . . . . . . X
X . . . . . . . . . . . . . X
X . . $ . . . . . . . . . . X
X X X X X X X X X X X X X X
```

Un momento, por favor...



Este es el listado de CAMPO MINADO:

```
   10 REM Campo minado
   20 GOTO250
   30 REM Mueve los robots
   40 CUENTA = 0
   50 FOR E=1 TO 4
   60 IF A(B(E),C(E)) = 42 THEN CUENTA = CUENTA +1
:GOTO 200
   70 X=B(E):Y=C(E)
   80 IF B(E)<D THEN B(E)=B(E)+1
   90 IF B(E)>D AND RND(-1)>.3 THEN B(E)=B(E)-1
  100 IF C(E)<F AND RND(-1)>.3 THEN C(E)=C(E)+1
```

```
110 IF C(E)>F THEN C(E)=C(E)-1
120 IF B(E)<2 THEN B(E)=2
130 IF B(E)>14 THEN B(E)=14
140 IF C(E)<2 THEN C(E)=2
150 IF C(E)>14 THEN C(E)=14
160 A(X,Y)=46
170 IF A(B(E),C(E)) = 42 THEN CUENTA = CUENTA +1
:GOTO200
180 IF A(B(E),C(E)) = 72 THEN A(B(E),C(E)) = 36:
GOSUB 300:GOTO 910
190 A(B(E),C(E)) = 36
200 NEXT E
210 IF CUENTA < CH THEN CUENTA = CH
220 CH= CUENTA
230 IF CUENTA = 4THEN GOSUB 300:GOTO 970
240 RETURN
250 GOSUB 530:REM Inicializacion
260 GOSUB 300:REM Imprime el campo
270 GOSUB 30:REM Movimiento de los robots"
280 GOSUB 420:REM Movimiento del humano
290 GOTO260
300 REM Imprime campo de minas
310 LOCATE0,0
320 IF CUENTA > 0 THEN PRINT"Cuenta de robots mu
ertos:"CUENTA:PRINT
330 IF CUENTA = 0 THEN PRINT:PRINT
340 PRINT:PRINT:PRINT
350 FOR B=1 TO 15
360 FOR C=1 TO 15
370 PRINT CHR$(A(B,C));" ";
380 NEXT C
390 PRINT
400 NEXT B
410 RETURN
420 REM Mueve el jugador
430 A(D,F) = 46
440 A$ = INKEY$
450 IF A$ = "" THEN 440
460 IF A$="N" AND D>2 THEN D=D-1
470 IF A$="S" AND D<14 THEN D=D+1
480 IF A$="E" AND F<14 THEN F=F+1
490 IF A$="O" AND F>2 THEN F=F-1
500 IF A(D,F)=42 THEN GOSUB 300:GOTO 990
```

```
510 A(D,F)=72
520 RETURN
530 REM Inicializacion
540 DEFINT A - Z
550 DIM A(15,15),B(4),C(4)
560 CLS
570 PRINT:PRINT"Un momento, por favor..."
580 CH = 0
590 RANDOMIZE
600 REM ****************************************
*************************
610 REM Situa las paredes
620 FOR B=1 TO 15
630 FOR C=1 TO 15
640 A(B,C) = 46
650 IF B=1 OR B= 15 OR C=1 OR C=15 THEN A(B,C) =
88
660 NEXTC:NEXT B
670 REM Situa las minas
680 FOR B=1 TO 20
690 C=INT(RND(-1)*13) + 1
700 D = INT(RND(-1)*13) + 1
710 IF A(C,D) =88 THEN 690
720 A(C,D) = 42
730 NEXT B
740 DATA 4,4,13,8,8,3,12,7
750 REM Situa los robots
760 FOR E=1 TO 4
770 D=INT(RND(-1)*13) + 2
780 F=INT(RND(-1)*13) + 2
790 IF A(D,F) <> 46 THEN 770
800 B(E)=D:C(E)=F
810 A(B(E),C(E))=36
820 NEXT E
830 REM Situa al humano
840 D=INT(RND(-1)*13)+2
850 F=INT(RND(-1)*13)+2
860 IF A(D,F) <> 46 THEN 830
870 A(D,F)=72
880 RETURN
890 REM Humano en d,f
900 REM Robots en b(e),c(e)
910 REM Final del juego
```

```
 920 CLS:PRINT"Un robot te ha alcanzado!!"
 930 CH=0:CUENTA=0:GOSUB 340
 940 A$=""
 950 IF INKEY$<>"" THEN 950
 960 GOSUB 570:GOTO260
 970 CLS:PRINT"Has vencido a los robots!!"
 980 GOTO930
 990 CLS:PRINT"Has chocado con una mina!!"
1000 GOTO 930
```

CEROS Y CRUCES

Estoy seguro de que este juego no necesita ninguna introducción. Este programa juega bastante bien, pero de un modo no del todo predecible, lo que hace que el jugador humano tenga alguna posibilidad de ganar de vez en cuando.

Puede que no suene muy justo, pero muchos programas de este juego están escritos de modo que el jugador humano, de no verse, de hecho, vencido por la máquina, a lo más que puede aspirar es a un empate.

Este programa decide quién abrirá el juego en cada partida y responde rápidamente a las jugadas del jugador humano. La manera de jugar de éste es simplemente introduciendo el número de la casilla a la que se quiere mover.

Vamos a echarle ahora una ojeada a una partida de CEROS Y CRUCES:

```
1 2 3   - - -        1 2 3   - - 0
4 5 6   - X -        4 5 6   - X -
7 8 9   - - -        7 8 9   - - -

                     Teclea tu movimiento
```

```
1 2 3   - - 0
4 5 6   - X -
7 8 9   - - X

Teclea tu movimiento
```

```
1 2 3   0 - 0
4 5 6   - X -
7 8 9   - - X

Teclea tu movimiento
```

```
1 2 3   0 X 0
4 5 6   - X -
7 8 9   - - X

Teclea tu movimiento
```

```
1 2 3   0 X 0
4 5 6   - X -
7 8 9   - 0 X

Teclea tu movimiento
```

```
1 2 3   0 X 0
4 5 6   X X -
7 8 9   - 0 X

Teclea tu movimiento
```

```
1 2 3   0 X 0
4 5 6   X X 0
7 8 9   - 0 X

Teclea tu movimiento
```

```
1 2 3   O X O
4 5 6   X X O
7 8 9   X O X

Tablas!
```

Y este es el listado del programa:

```
  10 REM Ceros y cruces
  20 DEFINT A-Z
  30 CLS
  40 N=0
  50 PRINT"Aprieta cualquier tecla cuando"
  60 PRINT"estes preparado para jugar"
  70 N=N+1: IF INKEY$="" THEN 70
  80 CLS
  90 DIM A(9)
 100 RANDOMIZE
 110 CLS: FOR N=1 TO 9:A(N)=0:NEXT N
 120 IF RND(1)>.5 THEN PRINT "Yo hago el primer m
ovimiento":FOR J=1 TO 1000:NEXT:CLS:GOTO 160
 130 GOSUB 750
 140 GOSUB 480
 150 GOSUB 670
 160 GOSUB 750
 170 GOSUB 480
 180 IF A(5)=0 THEN A(5)=1:GOTO 130
 190 REM Para completar la fila
 200 D=1
 210 B=1
 220 IF B=1 THEN X=1:Y=2:Z=3
 230 IF B=2 THEN X=1:Y=4:Z=7
 240 IF B=3 THEN X=1:Y=5:Z=9
```

```
250 IF B=4 THEN X=3:Z=7
260 C=1
270 IF A(X)=D AND A(Y)=D AND A(Z)=0 THEN A(Z)=1
: GOTO 130
280 IF A(X)=D AND A(Y)=0 AND A(Z)=D THEN A(Y)=1:
GOTO 130
290 IF A(X)=0 AND A(Y)=D AND A(Z)=D THEN A(X)=1:
 GOTO 130
300 IF B=1 THEN X=X+3:Y=Y+3:Z=Z+3
310 IF B=2 THEN X=X+1:Y=Y+1:Z=Z+1
320 IF C<3 THEN C=C+1: GOTO 270
330 IF B<4 THEN B=B+1: GOTO 230
340 IF D<2 THEN D=D+1: GOTO 210
350 REM Mueve aleatoriamente
360 B=1
370 D=INT (RND*9)+1
380 IF A(C)=0 THEN A(C)=1:GOTO 130
390 B=B+1
400 IF B<21 THEN 370
410 B=0
420 B=B+1
430 IF A(B)=0 THEN A(B)=1: GOTO 130
440 IF B<9 THEN 420
450 GOSUB 750
460 PRINT:PRINT "Es una linea"
470 GOTO 650
480 REM Comprobacion del ganador
490 FOR B=1 TO 4
500 IF B=1 THEN X=1:Y=2:Z=3
510 IF B=2 THEN X=1:Y=4:Z=7
520 IF B=3 THEN X=1:Y=5:Z=9
530 IF B=4 THEN X=3:Z=7
540 FOR C=1 TO 3
550 IF A(X)=A(Y) THEN IF A(Y)=A(Z) THEN IF A(X)<
>0 THEN 610
560 IF B=1 THEN X=X+3:Y=Y+3:Z=Z+3
570 IF B=2 THEN X=X+1:Y=Y+1:Z=Z+1
580 NEXT C
590 NEXT B
600 RETURN
610 REM El ganador
620 PRINT
630 IF A(X)=1 THEN PRINT "Soy el ganador!"
```

```
640 IF A(X)=2 THEN PRINT "Tu eres el ganador!"
650 FOR O=1 TO 1000:NEXT O
660 GOTO 110
670 REM Movimiento del jugador
680 PRINT: PRINT "Teclea tu movimiento"
690 A$=INKEY$
700 IF A$<"1" OR A$>"9" THEN 690
710 B= VAL(A$)
720 IF A(B)<>0 THEN 690
730 A(B)=2
740 RETURN
750 REM Impresion
760 CLS
770 PRINT:PRINT:PRINT:PRINT
780 PRINT"1 2 3   ";
790 FLAG=0
800 FOR B=1 TO 9
810 IF A(B)=0 THEN FLAG=1
820 IF A(B)=0 THEN PRINT " - ";
830 IF A(B)=1 THEN PRINT " O ";
840 IF A(B)=2 THEN PRINT " X ";
850 IF B=3 THEN PRINT: PRINT :PRINT "4 5 6   ";
860 IF B=6 THEN PRINT: PRINT: PRINT "7 8 9   ";
870 NEXT B
880 PRINT:PRINT
890 IF FLAG=0 THEN 460
900 RETURN
```

LAS VEGAS

Y para terminar este capítulo, tenemos el clásico juego de tragaperras, LAS VEGAS. Lo único que tiene que hacer el jugador es decidir cuánto quiere apostar antes de apretar la palanca del tragaperras (lo que en este caso se hace presionando el espaciador del ordenador) y las bobinas girarán a toda velocidad.

Las ganancias del jugador —como es fácil suponer— dependen de la relativa dificultad de las diferentes combinaciones que pueden aparecer. El ordenador no para de hacer ruido a medida que se va desarrollando el juego y el jugador se va enriqueciendo o empobreciendo.

Así es como se gana un millón (más o menos):

```
Hola, jugador!

Tienes $ 250

Cuanto quieres apostar?140


OK, senor, son $ 140


Aprieta la barra espaciadora para jugar
```

```
/^^^
```

```
/^^^
```

```
140 GOSUB 410
150 PRINT"Estas limpio, totalmente arruinado!!"
160 GOSUB 410
170 PRINT"Aprieta 'S' si te gustaria"
180 PRINT"intentar de nuevo  saltar la banca de"
190 PRINT"!!!!! LAS VEGAS !!!!!"
200 PRINT"(o oprime 'N' si quieres dejarlo)"
210 A$ = INKEY$
220 IF A$<>"S" AND A$<>"s" AND A$<>"N" AND A$<>"
n" THEN 210
230 IF A$="S" OR A$="s" THEN GOSUB 1150:GOTO30
240 PRINT:PRINT"OK, jugador..."
250 GOSUB 410
260 PRINT"Gracias por el juego!"
270 END
280 REM ******************************
290 REM        SALTA LA BANCA
300 REM ******************************
310 GOSUB 410
320 PRINT"Bien hecho, muchacho!!"
330 GOSUB 410
340 PRINT"Has alcanzado el limite de esta casa"
350 PRINT"asi que tendremos que echarte"
360 GOSUB 410
370 PRINT"Es la gente con suerte como tu la que
da"
380 PRINT"mala fama a nuestro casino.............
.."
390 GOTO160
400 REM ************************
410 REM          RETARDO
420 REM ************************
430 FOR P=1 TO 1000 :NEXT P
440 PRINT:PRINT
450 RETURN
460 REM ********************************
470 REM         SEGUNDO RETRASO
480 REM ********************************
490 FOR P= 1 TO 1000:NEXT P
500 RETURN
510 REM ***********************************
520 REM   MUEVE LA MAQUINA TRAGAPERRAS
530 REM ***********************************
```

```
540 CLS
550 GOSUB 410
560 PRINT"/^^^
```

```
  820 GOSUB 410
  830 IF CAJA > 0 THEN PRINT"Asi que ahora tienes
$"CAJA
  840 GOSUB 460
  850 RETURN
  860 REM ****************************
  870 REM    ENTRADA DEL JUGADOR
  880 REM ****************************
  890 CLS
  900 GOSUB 410
  910 IF CAJA < 300 THEN PRINT"Hola, jugador!"
  920 IF CAJA>299 AND CAJA<600 THEN PRINT"Lo estas
 haciendo bien esta noche..."
  930 IF CAJA > 599 AND CAJA<900 THEN PRINT"Cierta
mente, la Diosa de la Fortuna te ha sonreido!"
  940 IF CAJA > 899 AND CAJA<1200 THEN PRINT"La Fa
talidad parece haberse olvidado de ti"
  950 IF CAJA >1199 THEN PRINT"Puedo reconocer el
trabajo de un experto"
  960 GOSUB 470
  970 PRINT:PRINT"Tienes $"CAJA
  980 PRINT:INPUT"Cuanto quieres apostar";APUESTA
  990 IF APUESTA>CAJA THEN PRINT"No tienes tanto d
inero!":GOTO980
 1000 GOSUB 410
 1010 PRINT"OK, senor, son $"APUESTA
 1020 GOSUB 410
 1030 PRINT"Aprieta la barra espaciadora para juga
r"
 1040 IF INKEY$<>" " THEN 1040
 1050 FOR T= 1TO 10
 1060 PRINT TAB (T/2);"******* Un momento *******"
 1070 PRINT
 1080 NEXT T
 1090 RETURN
 1100 REM ****************************
 1110 REM        INICIALIZACION
 1120 REM ****************************
 1130 CLS
 1140 DIM A$(6),C(6)
 1150 RANDOMIZE
 1160 CAJA=250
 1170 FOR B=1 TO 6
```

```
1180 READ A$(B)
1190 NEXT B
1200 RETURN
1210 DATA"$ Manzana $","# Cereza #","* Campana *"
,"!! BARRA !!","<< Limon >>","{{ Ciruela }}"
```

Diversión con la impresora

CELESTIA

El primero de los dos programas que componen esta parte es CELESTIA, que el jugador puede ejecutar ya sea en la pantalla del televisor o en la pantalla del ordenador y la impresora al mismo tiempo. Produce una serie infinita de dibujos que evolucionan de diferentes maneras.

Los dibujos evolucionan de acuerdo con las reglas del famoso juego de ordenador LIFE que desarrolló John Conway durante su estancia en el Gonville and Caius College de Cambridge, en Gran Bretaña. Martin Gardiner divulgó este juego en el mundo entero a raíz de la publicación de un artículo suyo sobre el tema en el número de octubre de 1970, de la famosa revista *Scientific American.*

En LIFE las células nacen, se reproducen y mueren de acuerdo con las reglas inventadas por Conway. Una célula en una red (se imagina que la colonia de células se desarrolla en forma de red) está rodeada por ocho más y el estado de esas ocho dicta lo que pasará con la célula en cuestión en la siguiente generación.

Las reglas que gobiernan la evolución de las células son las siguientes:

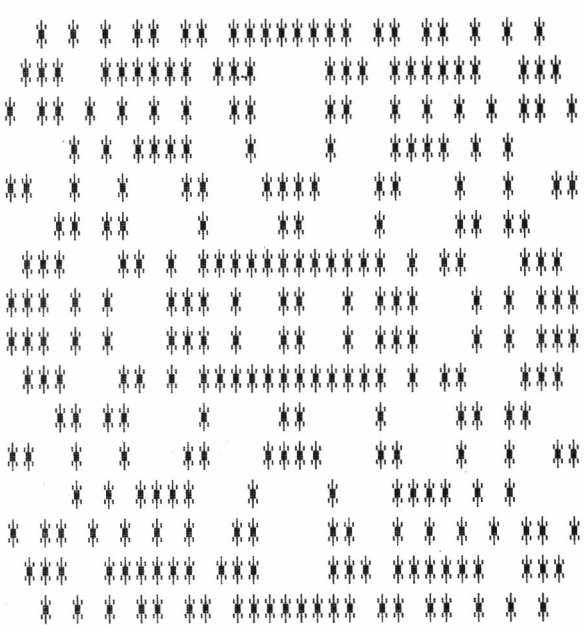
— Si a una célula la rodean dos o tres más, sobrevivirá en la siguiente generación.

— Si hay tres, y sólo tres, células llenas junto a una célula vacía, nacerá una célula nueva en ese espacio vacío en la siguiente generación.

— Una célula que tenga cuatro células vecinas llenas morirá en la siguiente generación.

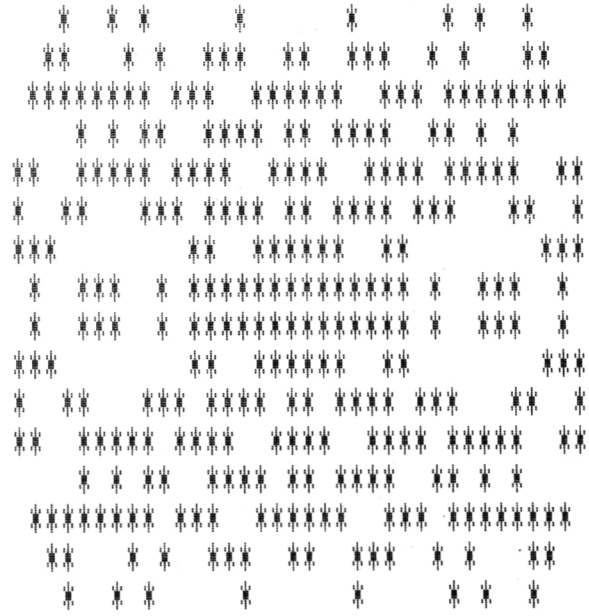
No se preocupe el lector pues no tendrá que aprenderse las reglas ya que el ordenador las interpreta bastante bien él solo. Estas reglas producen dibujos que son mucho más atractivos (y mucho menos predecibles) de lo que el lector podría pensar tras leer las reglas. Este programa sólo tiene un giro especial, lo que lo hace todavía más efectista, en relación con LIFE. En realidad, CELESTIA imprime cuatro colonias cada vez que una colonia evoluciona. La colonia original aparece en un cuarto de la pantalla y los otros tres cuartos de la pantalla contienen las derivaciones de la original.

CELESTIA es increíblemente efectista, como puede ver a continuación:



```
 **    **  ** ** ** ** **  **    **
  * ** * * **  **  **  ** * * ** *
*******   * **       ** *   *******
* * * ** * **** *  * **** * ** * * *
** ** ************************ ** **
 * * **    *** * ** * ***    ** * *
 *** * *** *   *    *   * *** * ***
 ***    ****   *    *   ****    ***
 ***    ****   *    *   ****    ***
 *** * *** *   *    *   * *** * ***
 * * **    *** * ** * ***    ** * *
** ** ************************ ** **
* * * ** * **** *  * **** * ** * * *
*******   * **       ** *   *******
  * ** * * **  **  **  ** * * ** *
 **    **  ** ** ** ** **  **    **
```

```
* *** *   ******    ******   * *** *
**        ** * **  ** * **        **
* *     * *  *  *  *  *  * *     * *
 ***  * ******  ****  ****** *  ***
      ***** * ** ** ** * *****
 * * **  *    * *  * *    *  ** * *
* * **   * ** * **** * ** *   ** * *
 * **      ***   **   ***      ** *
 * **      ***   **   ***      ** *
* * **   * ** * **** * ** *   ** * *
 * * **  *    * *  * *    *  ** * *
      ***** * ** ** ** * *****
 ***  * ******  ****  ****** *  ***
* *     * *  *  *  *  *  * *     * *
**        ** * **  ** * **        **
* *** *   ******    ******   * *** *
```



Este es el listado:

```
 10 REM Celestia
 20 GOSUB 450:REM Inicializacion
 30 GOSUB 60:REM Impresion
 40 GOSUB 280:REM Evolucion
 50 GOTO 30
 60 LOCATE 0,0
 70 PRINT:PRINT:PRINT ,
 80 FOR X=2 TO 9
 90 FOR Y=2 TO 19
100 PRINT CHR$(A(X,Y));
110 NEXT Y
120 FOR Y=19 TO 2 STEP -1
130 PRINT CHR$(A(X,Y));
140 NEXT Y
150 PRINT:PRINT,
160 NEXT X
170 FOR X=9 TO 2 STEP -1
180 FOR Y=2 TO 19
```

```
190 PRINT CHR$(A(X,Y));
200 NEXT Y
210 FOR Y=19 TO 2 STEP -1
220 PRINT CHR$(A(X,Y));
230 NEXT Y
240 PRINT:PRINT ,
250 NEXT X
260 IF N=0 THEN GOSUB 640
270 RETURN
280 FOR X=2 TO 9:FOR Y= 2 TO 19
290 C=0
300 IF A(X-1,Y-1) = B THEN C=C+1
310 IF A(X-1,Y) = B THEN C=C+1
320 IF A(X-1,Y+1) = B THEN C=C+1
330 IF A(X,Y-1) = B THEN C=C+1
340 IF A(X,Y+1) = B THEN C=C+1
350 IF A(X+1,Y-1) = B THEN C=C+1
360 IF A(X+1,Y) = B THEN C=C+1
370 IF A(X+1,Y+1) = B THEN C=C+1
380 IF A(X,Y) = B AND C<>3 ANDC<>2 THEN B(X,Y)=E
390 IF A(X,Y) <>E AND C=3 THEN B(X,Y)=B
400 NEXT Y:NEXT X
410 FOR X=2 TO 9:FOR Y=2 TO 19
420 A(X,Y)=B(X,Y)
430 NEXT Y:NEXT X
440 RETURN
450 REM Inicializacion
460 CLS
470 PRINT"Aprieta la barra espaciadora":PRINT"cu
ando estes preparado para empezar"
480 N=1
490 IF INKEY$="" THEN N=N+1:GOTO 490
500 RANDOMIZE:CLS
510 PRINT"Quieres una copia por"
520 PRINT"impresora (S o N)"
530 A$=INKEY$:IF A$ <> "S" AND A$<> "s" AND A$<>
 "N" AND A$<> "n" THEN 530
540 IF A$="S" OR A$="s" THEN N=0
550 CLS
560 B=ASC("*"):E=ASC(" ")
570 DIM A(10,20),B(10,20)
580 FOR X=2 TO 9:FOR Y= 2 TO 19
590 A(X,Y)=E
```

```
600 IF RND(-N)>.45 THEN A(X,Y) = B
610 B(X,Y)=A(X,Y)
620 NEXT Y:NEXT X
630 RETURN
640 LPRINT"                                    ----------
----":REM 6 espacios y 34 -'s
650 LPRINT:LPRINT:LPRINT"      ";
660 FOR X=2 TO 9
670 FOR Y=2 TO 19
680 LPRINT CHR$(A(X,Y));
690 NEXT Y
700 FOR Y=19 TO 2 STEP -1
710 LPRINTCHR$(A(X,Y));
720 NEXT Y
730 LPRINT:LPRINT"      ";
740 NEXT X
750 FOR X=9 TO 2 STEP -1
760 FOR Y=2 TO 19
770 LPRINT CHR$(A(X,Y));
780 NEXT Y
790 FOR Y=19 TO 2 STEO -1
800 LPRINT CHR$(A(X,Y));
810 NEXT Y
820 LPRINT:LPRINT"      "
830 NEXT X
840 LPRINT
850 RETURN
```

VALLA PUBLICITARIA

Este programa, como puede ver en el ejemplo que le ofrecemos a continuación, toma un mensaje de cualquier longitud (toda la que le permita la longitud de serie del sistema) y lo convierte en un mensaje impreso que se obtiene por la impresora. Las letras impresas pueden usarse entonces para decorar las paredes, como pósters, o como se quiera.

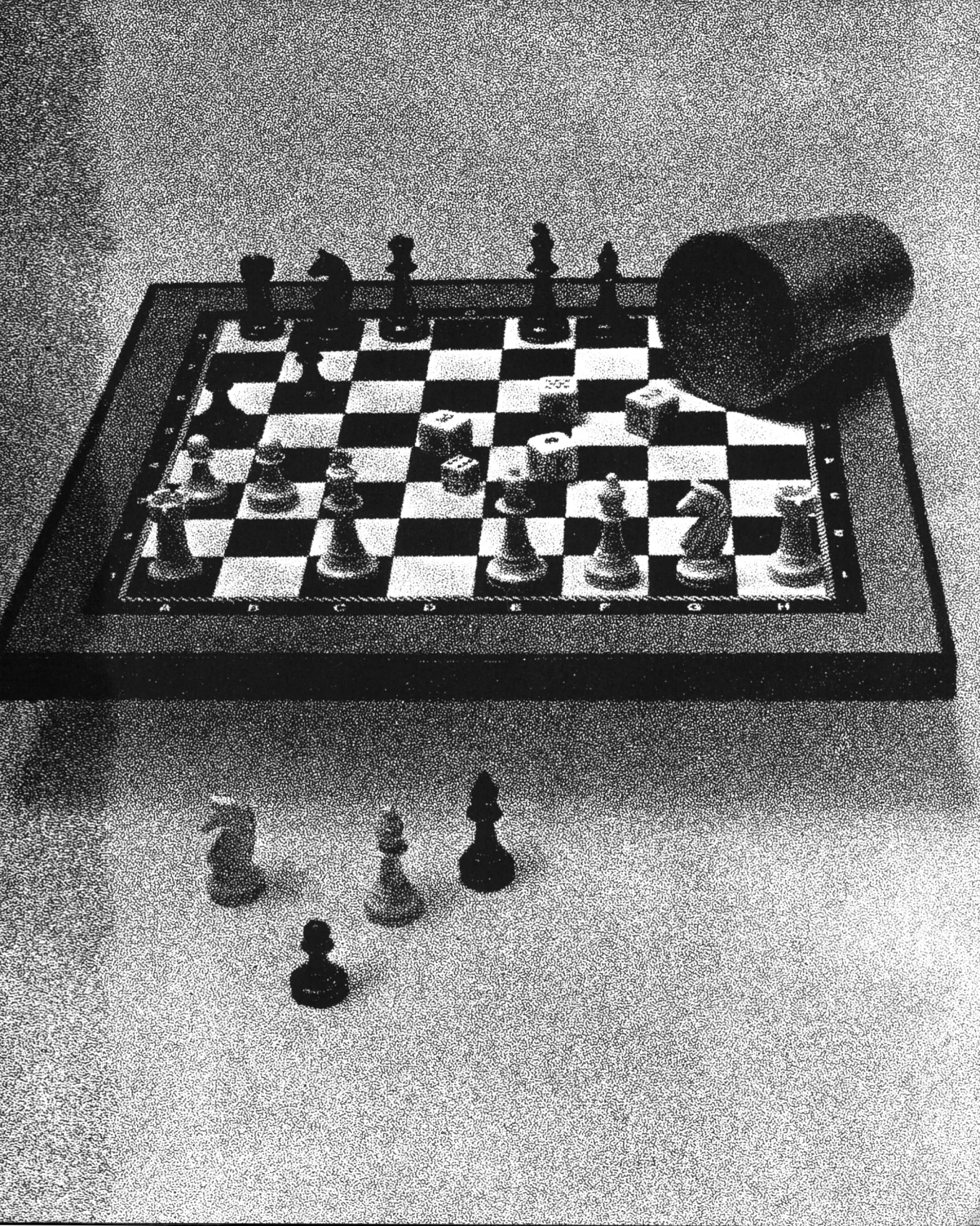
No es necesario dar muchas más explicaciones para conseguir un resultado como éste:

```
EEEEEEE
E  E  E
E  E  E
E     E
E     E

 S  SS
S S  S
S S  S
S S  S
SS   S

        T
        T
TTTTTTT
        T
        T

 OOOOO
O    O
O    O
O    O
 OOOOO

 S  SS
S S  S
S S  S
S S  S
SS   S

 OOOOO
O    O
O    O
O    O
 OOOOO

NNNNNNN
NN    N
NN    N
NN    N
NNNNNNN

LLLLLLL
L
L
L
L

EEEEEEE
E  E  E
E  E  E
E     E
E     E

        T
        T
TTTTTTT
        T
        T

RRRRRR
     R  R
     RR R
     R R R
     R   RR

AAAAA
A   A
A   A
AAAAA

NNNNNNN
NN    N
NN    N
NN    N
NNNNNNN

 GGGGG
G     G
G     G
G G   G
 GG  G

EEEEEEE
E  E  E
E  E  E
E     E
E     E

 S  SS
S S  S
S S  S
S S  S
SS   S

DDDDDDD
D     D
D     D
D     D
 DDDDD
```

Y este es el listado:

```
   10 REM Valla publicitaria
   20 REM Todas las entradas en mayusculas
   30 CLS
   40 INPUT"Cual es tu mensaje    ",A$
   50 B=LEN (A$)
   60 FOR C=1 TO B
   70 D= ASC(MID$(A$,C))-64
   80 IF D=-18 THEN GOSUB 410:GOTO 130:REM Punto (
.)
   90 IF D=-31 THEN GOSUB 420:GOTO 130:REM Exclama
cion (!)
  100 IF D=-32 THEN GOSUB 430:GOTO 130:REM Espacio
 ( )
  110 IF D<1 OR D>26 THEN GOSUB 430:GOTO 130:REM C
aracter desconocido
  120 ON D GOSUB 150,160,170,180,190,200,210,220,2
30,240,250,260,270,280,290,300,310,320,330,340,350
,360,370,380,390,400
  130 LPRINT:NEXT C
  140 END
  150 LPRINT "AAAAA":LPRINT "   A A":LPRINT"   A
A":LPRINT"   A A":LPRINT"AAAAA":RETURN
  160 LPRINT"BBBBBBB":LPRINT"B  B  B":LPRINT"B  B
 B":LPRINT"B  B  B":LPRINT" BB BB":RETURN
  170 LPRINT" CCCCC ":LPRINT"C     C":LPRINT"C
 C":LPRINT"C     C":LPRINT" C   C":RETURN
  180 LPRINT"DDDDDDD":LPRINT"D     D":LPRINT"D
 D":LPRINT"D     D":LPRINT" DDDDD":RETURN
  190 LPRINT"EEEEEEE":LPRINT"E  E  E":LPRINT"E  E
 E":LPRINT"E     E":LPRINT"E":RETURN
  200 LPRINT"FFFFFFF":LPRINT"   F  F":LPRINT"   F
 F":LPRINT"   F  F":LPRINT"      F":RETURN
  210 LPRINT" GGGGG":LPRINT"G     G":LPRINT"G
G":LPRINT"G G   G":LPRINT" GG  G":RETURN
  220 LPRINT"HHHHHHH":LPRINT"   H":LPRINT"   H":LP
RINT"   H":LPRINT"HHHHHHH":RETURN
  230 LPRINT:LPRINT"I      I":LPRINT"IIIIIII":LPRI
NT "I      I":LPRINT:RETURN
  240 LPRINT:LPRINT"JJJ":LPRINT"J     J":LPRINT"J
    J":LPRINT" JJJJJJ":RETURN
```

```
  250 LPRINT"KKKKKKK":LPRINT"   K":LPRINT"  K K":L
PRINT" K   K":LPRINT"K     K":RETURN
  260 LPRINT"LLLLLLL":FOR Z=1 TO 4:LPRINT"L":NEXT
Z:RETURN
  270 LPRINT"MMMMMMM":LPRINT"     M":LPRINT"   MM"
:LPRINT"     M":LPRINT"MMMMMMM":RETURN
  280 LPRINT"NNNNNNN":LPRINT"     NN":LPRINT"   NN
":LPRINT" NN":LPRINT"NNNNNNN":RETURN
  290 LPRINT" OOOOO":LPRINT"O     O":LPRINT"O
O":LPRINT"O     O":LPRINT" OOOOO":RETURN
  300 LPRINT"PPPPPP":LPRINT"   P  P":LPRINT"   P
P":LPRINT"   P  P":LPRINT"    PP":RETURN
  310 LPRINT" QQQQQ":LPRINT"Q     Q":LPRINT"Q Q
Q":LPRINT"QQ    Q":LPRINT" QQQQQ":RETURN
  320 LPRINT"RRRRRR":LPRINT"   R  R":LPRINT"  RR
R":LPRINT" R R  R":LPRINT"R   RR":RETURN
  330 LPRINT" S  SS":LPRINT"S  S  S":LPRINT"S  S
S":LPRINT"S  S  S":LPRINT" SS  S":RETURN
  340 LPRINT"      T":LPRINT"      T":LPRINT"TTTTT
TT":LPRINT"      T":LPRINT"      T":RETURN
  350 LPRINT" UUUUUU":LPRINT"U":LPRINT"U":LPRINT"U
":LPRINT" UUUUUU":RETURN
  360 LPRINT"   VVVV":LPRINT" VV":LPRINT"V":LPRINT
" VV":LPRINT"   VVVV":RETURN
  370 LPRINT" WWWWWW":LPRINT"W":LPRINT"WWWWW":LPRI
NT"W":LPRINT" WWWWWW":RETURN
  380 LPRINT"X     X":LPRINT" XX XX":LPRINT"   X":
LPRINT" XX XX":LPRINT"X     X":RETURN
  390 LPRINT"     YY":LPRINT"    Y":LPRINT"YYYY":L
PRINT"    Y":LPRINT"     YY":RETURN
  400 LPRINT"ZZ    Z":LPRINT"Z Z   Z":LPRINT"Z  Z
 Z":LPRINT"Z   Z Z":LPRINT"Z    ZZ":RETURN
  410 LPRINT"@@":LPRINT"@@":LPRINT:LPRINT:LPRINT:R
ETURN
  420 LPRINT"@@ @@@@":LPRINT"@@ @@@@":LPRINT:LPRIN
T:LPRINT:RETURN
  430 LPRINT:LPRINT:RETURN
```


A B C D E F G H

JUEGOS ESPACIALES

¡Ah!, el espacio, el antiguo e infinito espacio, la última frontera. Ninguna colección de juegos de ordenador sería una colección completa si no incluyera unos cuantos juegos espaciales. Lamentablemente, la mayoría de los juegos de este tipo se basan en el uso de conjuntos de caracteres gráficos particulares y en caracteres que se mueven. Por supuesto, si queríamos que esta colección de juegos fuera «universal», no podíamos asumir el acceso a un tipo especial de pantalla. Los PEEKS y POKES se encuentran entre los aspectos menos transportables de cualquier programa.

Así pues, los juegos espaciales de esta colección se basan ya sea sólo sobre el texto de salida, ya sea sobre los caracteres gráficos de salida de un tipo limitado (como usar <*> para representar la nave espacial). A pesar de la aparente limitación que esto supone, los programas funcionan sorprendentemente bien y, una vez que se consigue ejecutarlos, su adaptación, de modo que se puedan beneficiar de las ventajas de cada modelo de ordenador en particular, no debería resultar muy difícil.

ALUNIZAJE I y II

Hay dos versiones de este programa. Yo sugiero al lector que empiece por la más sencilla y cuando vea que consigue alunizar dos de cada tres veces, considere que ya puede pasar al segundo en el que se utilizan INKEY$ y que exige una gran agilidad de pensamiento por parte del jugador para salvar a su nave del desastre.

El escenario de estos programas es bastante familiar. El jugador está alunizando con poco combustible y por alguna razón el ordenador que lleva a bordo es incapaz de todo salvo de informarle sobre el estado del vuelo. La información que llega al jugador es limitada, pero adecuada.

El jugador está continuamente informado de la altura de vuelo, de la reserva de combustible y la velocidad de descenso y lo que ha de hacer es introducir el empuje correcto para caer lo más suavemente posible sobre la superficie de la luna. El combustible con que cuenta es limitado de modo que debe racionarlo con el fin de tener suficiente en los últimos segundos de vuelo para amortiguar el impacto. Una característica (líneas 370 y 380) asegura que el jugador y sus amigos volverán al programa una y otra vez para intentar conseguir un mejor «rango galáctico» y crear un cráter cada vez menos profundo.

Aquí puede ver ALUNIZAJE en acción:

```
Altura:  1220                Velocidad: -47
    Combustible:  355

        <*>

   Empuje?17

Tu nave se estrello a 28 k.p.h.
creando un crater de  1260 metros de profundidad

Tu puntuacion galactica es 72

La mejor puntuacion hasta ahora es  72

Por favor, aguarda tu proxima mision...
```

Y este es el listado para la versión más sencilla del juego:

```
 10 REM Alunizaje 1
 20 MP=-10000
 30 RANDOMIZE
 40 A=-20-INT(RND(-1)*60):REM Velocidad inicial
 50 B=1200+ INT(RND(-1)*280):REM Altura
 60 C=320 + INT(RND(-1)*90):REM Combustible
 70 CLS
 80 PRINT:PRINT:PRINT
 90 B=INT(B):A=INT(A):C=INT(C)
100 PRINT"Altura: "B,"Velocidad: "A
110 PRINT"    Combustible: "C
120 FOR Q=1 TO 16-B/100
130 PRINT
140 NEXT Q
150 PRINT TAB(5+ RND(-1)*3 -RND(-1)*3);"<*>"
160 FOR Q=16-B/100 TO 16
170 PRINT
180 NEXT Q
190 FOR P=1 TO 500:NEXT P
200 INPUT"Empuje";E
210 FOR P=1 TO 500:NEXT P
220 IF E>C THEN E=0
230 C=C-E
240 B=B+A+(T-5)/2
250 A=A+(T-5)/2
260 IF C<1 AND B>100 THEN 300
270 IF ABS (B) < 20 AND ABS(A) < 15 THEN PRINT"H
as aterrizado satisfactoriamente":PRINT:PRINT"Bien
 hecho, intrepido capitan...":PUNTUACION =C*234:GO
TO340
280 IF B>19 THEN 70
290 IF C>1 THEN 310
300 PRINT"Te has quedado sin combustible"
310 PRINT"Tu nave se estrello a"ABS(A)"k.p.h."
320 PRINT"creando un crater de "INT(ABS(A)*45)"m
etros de profundidad"
330 PUNTUACION = 100-ABS(A)
340 FOR P= 1TO 1000:NEXT P
350 PRINT:PRINT
```

```
360 PRINT"Tu puntuacion galactica es"PUNTUACION
370 IF PUNTUACION >MP THEN MP=PUNTUACION
380 PRINT:PRINT"La mejor puntuacion hasta ahora
es "MP
390 PRINT:PRINT"Por favor, aguarda tu proxima mi
sion..."
400 FOR P=1 TO 1500:NEXT P
410 GOTO 40
```

Esta versión es sólo para cabalgadores del espacio experimentados:

```
10 REM Aluoizaje 2
20 MP=-10000
30 RANDOMIZE
40 A=0
50 B=1200+ INT(RND(-1)*380):REM Altura
60 C=720 + INT(RND(-1)*90):REM Combustible
70 CLS
80 PRINT:PRINT:PRINT
90 B=INT(B):A=INT(A):C=INT(C)
110 PRINT"Altura: "B,"Velocidad: "C
120 PRINT"Combustible: "C
130 FOR Q=1 TO 16-B/100
140 PRINT
150 NEXT Q
160 PRINT TAB(15+ RND(-1)*3 -RND(-1)*3);"<*>"
170 FOR Q=16-B/100 TO 16
180 PRINT
190 NEXT Q
200 T=0
210 T$=INKEY$:T=VAL(T$)*1.7
220 IF E>C THEN E=0
230 C=C-E
240 B=B+A+(E-5)/2
250 A=A+(E-5)/2
260 IF C<1 AND B>100 THEN 300
270 IF ABS (B) < 40 AND ABS(A) < 25 THEN PRINT"H
as aterrizado satisfactoriamente":PRINT:PRINT"Bien
 hecho, intrepido capitan...":PUNTUACION =C*234:GO
TO340
```

```
280 IF B>39 THEN 70
290 IF C>1 THEN 310
300 PRINT"Te has quedado sin combustible"
310 PRINT"Tu nave se estrello a"ABS(A)"k.p.h."
320 PRINT"creando un crater de "INT(ABS(A)*45)"m
etros de profundidad"
330 PUNTUACION = 100-ABS(A)
340 FOR P= 1TO 1000:NEXT P
350 PRINT:PRINT
360 PRINT"Tu puntuacion galactica es"PUNTUACION
370 IF PUNTUACION >MP THEN MP=PUNTUACION
380 PRINT:PRINT"La mejor puntuacion hasta ahora
es "MP
390 PRINT:PRINT"Por favor, aguarda tu proxima mi
sion..."
400 FOR P=1 TO 1500:NEXT P
410 IF INKEY$<>"" THEN 410
420 GOTO 40
```

SUPERGUERRA

Este programa es una imitación espacial que pone al jugador al mando de una patrulla que vigila un sector del espacio en forma de cubo que mide diez parsecs por cada lado. Los enemigos son los Dosznty. Sólo hay una nave Dosznty en este sector, nave que se mueve lentamente a través del cubo durante todo el transcurso del juego. Aunque el jugador no puede destruir la nave, lo que sí puede hacer es matar Dosznty individuales. De hecho, el objetivo del juego es matar a todos los enemigos que sea posible antes de que se le agoten las reservas de energía. Aunque en este sector hay muchos más Doszntys que humanos, los enemigos son mucho peor tiradores y por tanto la posibilidad de que el jugador liquide a un buen montón de ellos antes de que termine su misión es bastante alta.

El jugador sólo puede disparar a los Dosznty cuando se encuentre a tres unidades de su posición. Intentar hacerlo cuando están fuera de alcance sólo sirve para que la tripulación responda "La nave Dosznty está fuera de rango".

El jugador cuenta con una cantidad limitada de energía y cada vez que sufre una derrota ésta disminuye.

No explicaré el juego con más detalle ya que el hacerlo sólo llevaría a disminuir el placer de jugarlo. El «comportamiento del espacio» no cambia de un juego a otro y por tanto el jugador encontrará que puede aprender a conocer los trucos de la guerra espacial desde la primera partida consiguiendo cada vez más puntos si continúa jugando. El juego tiende a explicarle las reglas por sí mismo a medida que avanza y el diseño de la pantalla está organizado de modo que refleje claramente lo que está sucediendo en cada momento.

Lo que sigue es un trozo de SUPERGUERRA en acción:

```
                    Informe General del Control Central

Energia en los reactores principal y secundarios: 64
                El medidor interestelar de tiempo indica 50
*************************************************************

>>> Las coordenadas galacticas de la nave son 10  7  9

La nave nodriza Dosznty localizada en 1  2  4

Los Dosznty estan hacia el norte oeste enfrente nuestro, senor
*************************************************************



Cual es su orden capitan?
   N, S, E, O
   A(vanzar), R(etroceder)
   H(iperespacio)
   L(aser)
?L


          La nave de los Dosznty esta fuera de rango


          ****************************************
          * Los scaners de largo alcance indican: *
          *      N/S --2                         *
          *      E/O --1                         *
          *      A/R --5                         *
          ****************************************
```

Alerta Roja!, Alerta Roja!

Los Dosznty nos estan disparando!

Alerta Roja!, Alerta Roja!

Los disparos enemigos nos han alcanzado, capitan

El Control Central informa unos danos de 9
a la camara de reserva de potencia, capitan

Vencistes a 234563 Dosznty

Los sistemas de vida se agotan...

agotan...

agotan...

La cuenta ha sido 0

El mejor hasta ahora es 0

Aguarda tu proxima mision

Este programa permitirá que la consola de su ordenador se convierta en la consola del ordenador de una nave espacial cada vez que el jugador se proponga liberar el cubo de diez parsecs de la amenaza de la nave nodriza de los Doszntу:

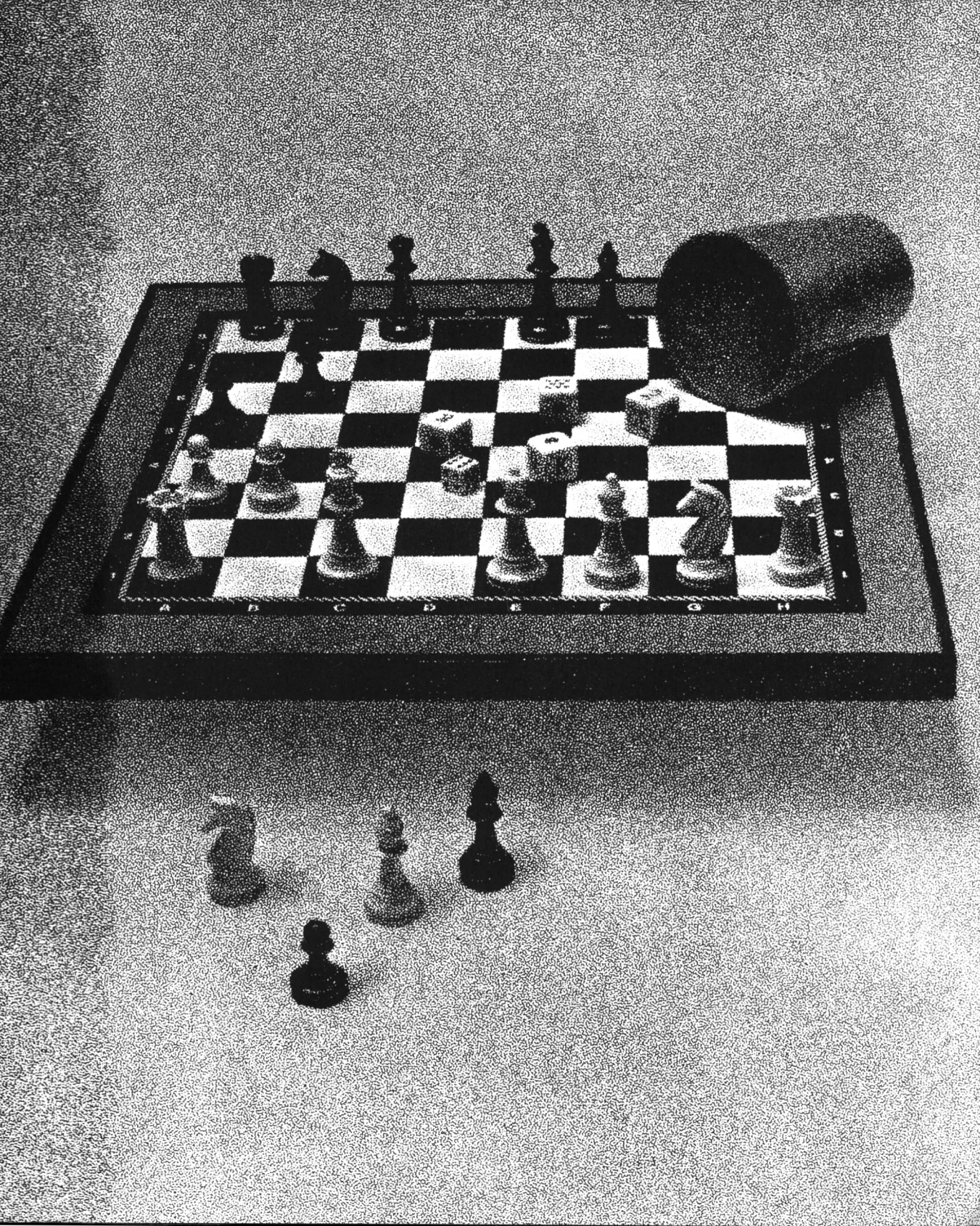
```
 10 REM Superguerra
 20 MP=0
 30 GOSUB 1410
 40 GOSUB 1150
 50 IF L<0 THEN 670
 60 GOSUB 1660
 70 L=L-.25
 80 TI=TI-1
 90 PRINT"Cual es su orden capitan?"
100 PRINT"    N, S, E, O"
110 PRINT"    A(vanzar), R(etroceder)"
120 PRINT"    H(iperespacio)"
130 PRINT"    L(aser)"
140 INPUT Z$
150 IF Z$="N" OR Z$="n" THEN X=X-1
160 IF Z$="S" OR Z$="s" THEN X=X+1
170 IF Z$="E" OR Z$="e" THEN Y=Y+1
180 IF Z$="O" OR z$="o" THEN Y=Y-1
190 IF Z$="A" OR Z$="a" THEN Z=Z-1
200 IF Z$="R" OR Z$="r" THEN Z=Z+1
210 IF Z$="L" OR Z$="l" THEN GOSUB 490
220 IF Z$="H" OR Z$="h" THEN GOSUB 1520
230 IF X<1 THEN X=1
240 IF Y<1 THEN Y=1
250 IF Z<1 THEN Z=1
260 IF X>10 THEN X=10
270 IF Y>10 THEN Y=10
280 IF Z>10 THEN Z=10
290 GOSUB 890
300 IF RND(-1)<.7 THEN 40
310 PRINT"*****************************************
********"
320 PRINT"* Los scaners de largo alcance indican
: *"
330 PRINT"*       N/S -"A-X"
      *"
```

```
340 PRINT"*       E/O -"B-Y"
     *"
350 PRINT"*       A/R -"C-Z"
     *"
360 PRINT"*************************************
***"
370 A=A+INT(RND(-1)*2 - RND(-1)*2)
380 B=B+INT(RND(-1)*2 - RND(-1)*2)
390 C=C+INT(RND(-1)*2 - RND(-1)*2)
400 IF A<1 THEN A=1
410 IF A> 10 THEN A=10
420 IF B<1 THEN B=1
430 IF B>10 THEN B=10
440 IF C<1 THEN C=1
450 IF C>10 THEN C=10
460 GOSUB 1650
470 GOTO40
480 REM ****************************************
************************
490 REM Opcion Laser
500 L=L-.75
510 GOSUB 1620
520 IF ABS(A-X) > 3 OR ABS(B-Y)>3 OR ABS(C-Z)>3
THEN PRINT"La nave de los Doszny esta fuera de ra
ngo"
530 IF ABS(A-X)>3 OR ABS(B-Y)>3 OR ABS(C-Z)>3 TH
EN GOSUB 1650:RETURN
540 PRINT"Lasers montados y dispuestos, capitan"
550 GOSUB 1650
560 IF RND(-1)>.5 THEN 590
570 PRINT"El disparo no ha tenido exito..."
580 GOTO650
590 PRINT"Hemos danado la nave Doszny, capitan!
"
600 FOR J=1 TO 30
610 FOR H=1 TO 30-J/3:PRINT" ";:NEXT H
620 PRINT"Bien hecho!!!";
630 NEXT J
640 T=T + INT(RND(-1)*1000)+784
650 GOSUB 1620
660 RETURN
670 REM Sin energia
680 GOSUB 1620
```

```
  690 PRINT"Este es el Control Central de la nave"
  700 GOSUB 1650
  710 IF L<= 0 THEN PRINT "Las reservas de energia
 estan agotadas"
  720 IF TI=0 THEN PRINT"Has permanecido en el esp
acio demasiado tiempo"
  730 PRINT"Vencistes a"T"Doszntу":GOSUB 1650
  740 PRINT"Los sistemas de vida se agotan..."
  750 GOSUB 1650
  760 PRINT"       agotan..."
  770 GOSUB 1650
  780 PRINT"            agotan..."
  790 GOSUB 1650
  800 PRINT"La cuenta ha sido "T:IF T>MP THEN MP=T
  810 PRINT:PRINT"El mejor hasta ahora es"MP:GOSUB
 1650
  820 PRINT:PRINT"Aguarda tu proxima mision"
  830 GOSUB 1650:GOSUB 1650:GOTO30
  840 REM ****************************************
***************************
  850 GOSUB 1650
  860 PRINT"Has colisionado con la nave de los Dos
znty!!!"
  870 GOSUB 1650
  880 GOTO690
  890 REM Informe Doszntу
  900 IF ABS(A-X)>5 OR ABS(B-Y)>5 OR ABS(C-Z)>5 OR
 RND(-1)<.7 THEN RETURN
  910 GOSUB 1620
  920 PRINT"Alerta Roja!,    Alerta Roja!"
  930 GOSUB 1650:IF RND(-1)<.25 THEN GOSUB 1620:PR
INT"Alerta Amarilla!":GOSUB1650:RETURN
  940 PRINT"Los Doszntу nos estan disparando!"
  950 GOSUB 1650
  960 PRINT"Alerta Roja!,   Alerta Roja!"
  970 GOSUB 1650
  980 IF RND(-1)>.7 THEN 1110
  990 PRINT"Los disparos enemigos nos han alcanzad
o, capitan"
 1000 DA = INT(RND(-1)*9)+2
 1010 GOSUB 1650
 1020 PRINT"El Control Central informa unos danos
de"DA
```

```
1030 PRINT"a la ";
1040 IF DA=2 THEN PRINT"seccion delanterá, senor"
1050 IF DA>2 AND DA<4 THEN PRINT"seccion de la tr
ipulacion, senor"
1060 IF DA>3 AND DA<7 THEN PRINT"turbina principa
l, senor"
1070 IF DA > 6 THEN PRINT"camara de reserva de po
tencia, capitan"
1080 L=L-DA
1090 GOSUB 1620
1100 RETURN
1110 GOSUB 1650
1120 PRINT"El disparo de los Doszntу no nos ha al
canzado, senor"
1130 GOSUB 1650
1140 RETURN
1150 REM Informe general
1160 CLS
1170 PRINT"*************************************
*****************************************"
1180 PRINT:PRINT"                    Informe General
del Control Central"
1190 GOSUB 1650
1200 PRINT"Energia en los reactores principal y s
ecundarios:"L
1210 IF L<3 THEN PRINT:PRINT"Cuidado... el nivel
de energia":PRINT"esta peligrosamente bajo!"
1220 PRINT,"El medidor interestelar de tiempo ind
ica"TI
1230 IF TI<8 THEN PRINT"!!!!!!Cuidado - Se agota
el tiempo para la mision!!!!!!!"
1240 IF TI<1 THEN 680
1250 PRINT"*************************************
*****************************************"
1260 IF T>0 THEN PRINT:PRINT"La cuenta de destruc
cion de Doszntys es"T
1270 PRINT">>> Las coordenadas galacticas de la n
ave son"X;Y;Z
1280 PRINT:PRINT"La nave nodriza Doszntу localiza
da en"A;B;C
1290 PRINT:PRINT"Los Doszntу estan ";
1300 IF A<>X OR B<>Y THEN PRINT"hacia el";
1310 IF A<X THEN PRINT" norte";
```

```
1320 IF A>X THEN PRINT" sur";
1330 IF B>Y THEN PRINT" ESTE";
1340 IF B<Y THEN PRINT" oeste";
1350 IF C=Z THEN PRINT" de la nave, senor"
1360 IF C>Z THEN PRINT" detras de nosotros, senor
"
1370 IF C<Z THEN PRINT" enfrente nuestro, senor"
1380 PRINT"*************************************
*****************************************"
1390 RETURN
1400 REM ****************************************
*******************
1410 REM Inicializacion
1420 CLS
1430 RANDOMIZE
1440 L=35+INT(RND(-1)*30):T=0:TI=50
1450 A= INT(RND(-1)*10)+1
1460 B= INT(RND(-1)*10)+1
1470 C= INT(RND(-1)*10)+1
1480 X= INT(RND(-1)*10)+1
1490 Y= INT(RND(-1)*10)+1
1500 Z= INT(RND(-1)*10)+1
1520 REM Opcion Hiperespacio
1530 X=INT(RND(-1)*10)+1
1540 Y=INT(RND(-1)*10)+1
1550 Z=INT(RND(-1)*10)+1
1560 FOR J=1 TO 10
1570 PRINT TAB(J);"*":PRINT
1580 FOR H=1 TO J:NEXT H
1590 NEXT J
1600 CLS
1610 RETURN
1620 FOR P=1 TO 1000:NEXT P
1630 CLS
1640 GOTO 1660
1650 FOR P=1 TO 1000:NEXT P
1660 PRINT:PRINT
1670 RETURN
```


A B C D E F G H

JUEGOS MENTALES

Supongo que un 90 por 100 de los juegos podrían, sin estirar demasiado esta definición, denominarse «juegos mentales». Después de todo, todos ellos exigen una cierta cantidad de poder cerebral y de aplicación por parte del jugador.

Sin embargo, los programas que se incluyen en esta parte exigen un poco más de esfuerzo intelectual que algunos de los otros programas que aparecen en este libro.

Encontrará aquí el FASTERMIND en el que tendrá que deducir un código de cuatro números inventado por el ordenador; asimismo tendrá que comprobar su habilidad matemática en IDAHO STARS; y salvar el pellejo deduciendo la palabra escogida por el ordenador en EJECUTOR.

Asimismo, tendrá que desbloquear un retorcido cubo en CUBICO, se calmará con una partida o dos de canicas en MUMBLE MARBLE (un juego que normalmente, en la «vida real», se conoce con el nombre de «Solitaire») y tendrá que trabajar bajo la presión de un incansable reloj para localizar unos átomos en CYCLOTRON.

Si a su cerebro le queda todavía algo de fuerza, puede intentar una tarea que puede parecer imposible en FLIPPER y calcular en el menor número de jugadas posible, un algoritmo para solucionar el problema que plantea SWITCHEROO.

Finalmente, cuando su cerebro ya esté totalmente en forma después de solucionar todos estos problemas, puede intentar deducir el mapa de un sistema de cavernas en EN BUSCA DE DARYL. No se imagine que este es un juego fácil. Es necesario un alto grado de concentración y paciencia.

FASTERMIND

Empezaremos con algo bastante sencillo: FASTERMIND. Es posible que el lector esté totalmente familiarizado con este tipo de juegos en los que uno de los jugadores intenta romper el código desarrollado por el otro.

En este programa, el ordenador piensa en un código de cuatro números (utilizando los números de 1 al 9 y asegurándose de que no se repite ningún número dentro del código) y el jugador ha de intentar calcular cuál es ese código en no más de ocho jugadas.

Después de cada intento por parte del jugador humano, el ordenador responde en términos de «blancos» y «negros», siendo un «negro» la recompensa que consigue el jugador cada vez que dice un número correcto en la posición correcta y un «blanco» cuando dice un número correcto, pero que no está en la posición correcta dentro del código.

Este es el listado, así que ya puede poner a trabajar a su cerebro:

```
 10 REM Fastermind
 20 CLS
 30 N=1
 40 PRINT"Aprieta una tecla"
 50 N=N+1
 60 IF INKEY$="" THEN 50
 70 RANDOMIZE
 80 CLS
 90 PRINT"Fastermind"
100 PRINT
110 PRINT"Cuando yo te diga"
```

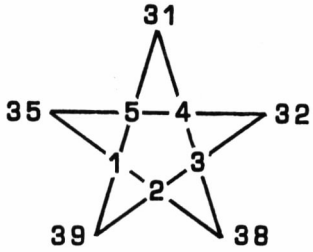
```
120 PRINT"escribe un numero de 4 cifras"
130 PRINT"y luego aprieta RETURN."
140 PRINT
150 PRINT"Los digitos pueden estar repetidos."
160 PRINT
170 PRINT"Tienes 8 posibilidades para romper"
180 PRINT"el dificil codigo."
190 FOR Z=1 TO 3000:NEXT Z
200 CLS
210 DIMB(4)
220 DIM D(4)
230 H=0
240 FOR A=1 TO 4
250 B(A)=INT(RND(-N)*9)+1
260 NEXT A
270 FOR C=1 TO 8
280 PRINT
290 PRINT"Dime tu intento";C;
300 INPUT X
310 IF X>9999 THEN 290
320 IF X<1000 THEN 290
330 P=INT(X/1000)
340 Q=INT((X-1000*P)/100)
350 R=INT((X-1000*P-100*Q)/10)
360 S=INT(X-1000*P-100*Q-10*R)
370 D(1)=P
380 D(2)=Q
390 D(3)=R
400 D(4)=S
410 FOR E=1 TO 4
420 IF D(E)<>B(E) THEN 470
430 PRINT" Negro";
440 B(E)=B(E)+10
450 D(E)=D(E)+20
460 H=H+1
470 NEXT E
480 IF H=4 THEN 680
490 FOR F=1 TO 4
500 D=D(F)
510 FOR G=1 TO 4
520 IF D<>B(G) THEN GOTO 560
530 PRINT"  Blanco";
540 B(G)=B(G)+10
```

```
550 GOTO 570
560 NEXT G
570 NEXT F
580 FOR G=1 TO 4
590 IF B(G)<10 THEN GOTO 610
600 B(G)=B(G)-10
610 NEXT G
620 H=0
630 PRINT
640 NEXT C
650 PRINT:PRINT"No lo conseguistes..."
660 PRINT"La respuesta es:   ";B(1);B(2);B(3);B(
4)
670 END
680 PRINT:PRINT:PRINT"Bien hecho, Cerebro Privil
egiado!"
690 PRINT
700 PRINT:PRINT"Conseguistes la respuesta en"
710 PRINT TAB(5);"solo";C;"intentos"
```

ESTRELLAS DE IDAHO

Las ESTRELLAS DE IDAHO tienen una propiedad especial: cuando se suman los números situados en los ángulos de las puntas que las forman, saldrá siempre el mismo total, independientemente de qué línea de números se escoja.

Esta, por ejemplo, es una de esas estrellas:



Como se puede ver, cada línea de la estrella suma lo mismo. En este programa, el ordenador genera una estrella de este tipo dejando en cero dos o tres de los números. La tarea del jugador es calcular cuáles son esos números en el menor espacio de tiempo.

Lo que sigue es un ejemplo del programa en acción:

```
                    2

0   12   17   1

    8      11

       10

0          0

Tienes 7 por ahora

Este es el intento No 2

Dime cualquier numero que
creas que es parte de la estrella?12

          2

12   12   17   1

    8      11

       10

0          12

Tienes 9 por ahora

Este es el intento No 3

Dime cualquier numero que
creas que es parte de la estrella?20
```

```
          2

   12   12   17   1

      8      11

         10

    20          12


Si, lo has resuelto!

y solo necesitastes 3 intentos...
Oprime 'S' si quieres
intentar otra estrella,
oprime 'N' para terminar...
```

Este es el listado que le permitirá producir sus propias estrellas (y observe que puede cambiar el 3 de la línea 630 por un número más alto si quiere producir estrellas más difíciles de resolver):

```
   10 REM Estrellas de Idaho
   20 GOSUB 480:REM Prepara la estrella
   30 GOSUB 340:REM Imprime la estrella
   40 GOSUB 70:REM Pide intento
   50 GOTO30
   60 REM ****************************************
**********************
   70 REM Pide un intento
   80 IF PUNTUACION > 0 AND PUNTUACION < 10 THEN P
RINT"Tienes"PUNTUACION"por ahora":PRINT
   90 VEZ=VEZ+1
  100 PRINT"Este es el intento No"VEZ:PRINT
```

```
110 PRINT"Dime cualquier numero que"
120 INPUT"creas que es parte de la estrella";G
130 PUNTUACION=0
140 FOR J=1 TO 10
150 IF G=A(J) THEN B(J) = A(J)
160 IF B(J) <>0 THEN PUNTUACION = PUNTUACION + 1
170 NEXT J
180 FOR P=1 TO 1000:NEXT P
190 IF PUNTUACION <10 THEN RETURN
200 REM ****************************************
*****************
210 GOSUB 340
220 PRINT:PRINT"Si, lo has resuelto!"
230 PRINT:PRINT"y solo necesitastes"VEZ"intentos
..."
240 FOR P=1 TO 1000:NEXT P
250 PRINT"Oprime 'S' si quieres"
260 PRINT"intentar otra estrella,"
270 PRINT"oprime 'N' para terminar..."
280 A$=INKEY$
290 IF A$<>"N" AND A$<>"n" AND A$<>"S" AND A$<>"
s" THEN 280
300 IF A$="S" OR A$="s" THEN GOSUB 530:RETURN
310 PRINT:PRINT"OK, gracias por jugar"
320 END
330 REM ****************************************
*****************
340 REM Imprime la estrella
350 CLS:PRINT:PRINT
360 PRINT TAB(11);B(1)
370 PRINT:PRINT
380 PRINT TAB(2);" ";B(2);" ";B(3);" ";B(4);" "B
(5)
390 PRINT
400 PRINTTAB(7);B(6);"    ";B(7)
410 PRINT
420 PRINTTAB(10);B(8)
430 PRINT
440 PRINTTAB(4);B(9);"        ";B(10)
450 PRINT:PRINT
460 RETURN
470 REM ****************************************
*******************
```

```
480 REM Prepara la estrella
490 CLS
500 RANDOMIZE
510 DIMA(10),B(10)
520 DEFINTA - Z
530 VEZ=0:PUNTUACION = 0
540 A=INT(RND(-1)*9)+1
550 B=INT(RND(-1)*9)+1
560 C=INT(RND(-1)*9)+1
570 D=INT(RND(-1)*9)+1
580 E=INT(RND(-1)*9)+1
590 IF A=B OR A=C OR A=D OR A=E THEN 550
600 IF B=C OR B=D OR B=E THEN 550
610 IF C=D OR C=E THEN 550
620 IF D=E THEN 550
630 X=INT(RND(-1)*3)+1
640 A(1)=X
650 A(2)=X-B+C+D
660 A(3)=A+E
670 A(4)=A+D
680 A(5)=X-B-C+E
690 A(6)=A
700 A(7)=A+C
710 A(8)=A+B
720 A(9)=X - 2*B +2*D + E
730 A(10) = X-2*B -C + D +2*E
740 FOR J=1 TO 10
750 B(J)=A(J)
760 IF A(J)=0 THEN GOSUB 530:RETURN
770 NEXT J
780 B(INT(RND(-1)*10)+1)=0
790 B(INT(RND(-1)*10)+1)=0
800 B(INT(RND(-1)*10)+1)=0
810 RETURN
```

EJECUTOR

Como verá en seguida, este juego es una versión para ordenador del conocido juego infantil LA HORCA. En este juego el ordenador escoge una palabra del banco de palabras que lleva incluido (contenido en las instrucciones DATA de las líneas comprendidas entre la 500 y la 540) y después reta al jugador humano a que la adivine.

El número de oportunidades con que cuenta el jugador depende de la longitud de la palabra (ver líneas 120 y 170).

Veamos ahora a un humano (yo) intentando adivinar la palabra que ha escogido el ordenador:

```
........
Tienes 0 letras correctas

                                  .E...A..
Te quedan 10 oportunidades...
                                  Tienes 2 letras correctas
Dime tu siguiente letra?E

                                  Te quedan 10 oportunidades...

                                  Dime tu siguiente letra?R
```

```
RE...A..                          RETOCA.O

Tienes 3 letras correctas         Tienes 7 letras correctas


Te quedan 10 oportunidades...     Te quedan 3 oportunidades..

Dime tu siguiente letra?D         Dime tu siguiente letra?D

          Hey! Te has librado de
          la ejecucion por otro dia
          Lo conseguistes en 8 intentos

          La palabra del Ejecutor era RETOCADO
```

Este listado le permitirá arriesgar su pellejo: la noble causa de incrementar su vocabulario.

```
 10 REM Ejecutor
 20 CLS:CLEAR
 30 N=0:Y=0
 40 PRINT"Aprieta 'E' cuando estes dispuesto"
 50 PRINT"a afrontar al Ejecutor..."
 60 N=N+1:IF INKEY$<>"E" AND INKEY$<>"e" THEN 60
 70 RANDOMIZE
 80 FOR G=1 TO RND(-N)*22+1
 90 READ A$
100 NEXT G
110 GOSUB 480
120 N=LEN(A$):DIMB(N),D(N)
130 FOR G=1 TO N
140 B(G) = ASC(MID$(A$,G,1))
150 D(G)=B(G)
160 NEXT G
```

```
170 Q=INT(N+N/4 + .5)
180 CLS
190 PRINT:PRINT"Tienes que acertar la palabra de
l"
200 PRINT"Ejecutor en solo"Q"intentos
205 FOR Z=1 TO 2000:NEXT
210 FOR J=1 TO Q:Y=Y+1
220 GOSUB 400
230 IF H=N THEN 340
240 PRINT:PRINT:PRINT"Te quedan"Q+1-J"oportunida
des..."
250 PRINT:INPUT"Dime tu siguiente letra";C$
260 F=ASC(C$)
270 FOR G=1 TO N
280 IF D(G)=F THEN D(G)=0:J=J-1
290 NEXTG:NEXTJ
300 GOSUB 480
310 GOSUB 400
320 PRINT:PRINT:PRINT"Lo siento, pero tu hora ha
 llegado!"
330 GOTO370
340 PRINT:PRINT"Hey! Te has librado de"
350 PRINT"la ejecucion por otro dia"
360 PRINT"Lo conseguistes en"Y-1"intentos"
370 PRINT:PRINT"La palabra del Ejecutor era ";A$
380 END
400 H=0:CLS:PRINT:PRINT:PRINT
410 FOR E=1 TO N
420 IF B(E)=D(E) THEN PRINT".";
430 IF B(E)<>D(E) THEN PRINT CHR$(B(E));:H=H+1
440 NEXT E
450 PRINT:PRINT:IF H<>N THEN PRINT"Tienes"H"letr
as correctas"
460 PRINT
470 RETURN
480 FOR O=1 TO 500:NEXT O
490 RETURN
500 DATA"MERIDIANO","MERITO","MERMADO","MERITOSO
"
510 DATA"OBISPO","OXIDANTE","OXIGENO","PALPABLE"
,"HETERODOXO"
520 DATA"PANDEMONIUM","PANEGIRICO","PARADOJICO",
"POBLADO"
```

```
  530 DATA"RAMPA","REMOLINO","SACRAMENTO","SABLE",
"ESQUEMATICO"
  540 DATA"SEDIMENTO","SEXAGENARIO","TEMPLADO","TE
LESCOPIO"
```

CUBISTA

Esta es una versión en dos dimensiones del famoso y colorido cubo que hace algún tiempo invadió el mundo desde Hungría.

El cubo empieza con este aspecto:

1	1	2	2
1	1	2	2
3	3	4	4
3	3	4	4

Después de cierta manipulación con el ordenador, tendrá este aspecto:

1	2	1	2
3	3	3	2
3	1	4	1
4	2	4	4

La labor del jugador es volverlo a poner tal como estaba al principio en el menor número de jugadas posible. Los números del cubo rotan en número de cuatro girando en la dirección de las agujas del reloj. Esto significa que:

1	2
3	4

al girarlos tendrán este aspecto:

```
3  1
4  2
```

Para rotar los elementos del cubo se introduce un número. La clave para mover es la siguiente:

```
 2   3   4
 6   7   8
10  11  12
```

Veamos ahora qué efecto tienen un par de rotaciones en el cubo que hemos visto un poco más arriba:

```
3  2  1  2
1  2  1  2
4  1  3  4
3  4  3  4

Que giro quieres hacer (2 - 12),menos
el 5 o el 9?6

3  2  1  2
4  1  1  2
1  2  3  4
3  4  3  4

Tu giro cubista numero 1

Que giro quieres hacer (2 - 12),menos
el 5 o el 9? 6
```

```
3  2  1  2
1  4  1  2
2  1  3  4
3  4  3  4


Tu giro cubista numero 2

Que giro quieres hacer (2 - 12),menos
el 5 o el 9?
```

Este es el listado:

```
 10 REM Cubista
 15 CLS:CLEAR
 20 DIM A(4),B(16)
 30 RANDOMIZE
 40 REM
 50 B=1:F=0
 60 FOR D=1 TO 4
 70 C=0
 80 IF B=1 THEN C=1
 90 IF B=3 THEN C=2
100 IF B=9 THEN C=3
110 IF B=11 THEN C=4
120 B(B)=C
130 B(B+1)=C
140 B(B+4)=C
150 B(B+5)=C
160 C=0
170 IF B=1 THEN C=2
180 IF B=3 THEN C=6
190 IF B=9 THEN C=2
200 B=B+C
210 NEXT D
220 GOSUB 410
```

```
230 PRINT:PRINT
240 IF F<11 THEN PRINT"Girando..."
250 F=F+1
260 FOR P=1 TO 900:NEXT P
270 IF F<11 THEN X=INT(RND(-1)*12)+1
280 IF F>10 THEN PRINT"Que giro quieres hacer (2
- 12),menos":INPUT"el 5 o el 9";X
290 IF X<2 OR X=5 OR X=9 OR X>12 THEN 270
300 A(1)=B(X)
310 A(2)=B(X+4)
320 A(3)=B(X+3)
330 A(4)=B(X-1)
340 B(X)=A(4)
350 B(X+4)=A(1)
360 B(X+3)=A(2)
370 B(X-1)=A(3)
380 GOSUB 410
390 F=F+1
400 GOTO270
410 CLS
420 PRINT:PRINT:PRINT
430 FOR B=1 TO 16
440 PRINTB(B);
450 IF B/4 = INT(B/4) THEN PRINT
460 NEXT B
470 PRINT:PRINT
480 IF F<10 THEN PRINT:PRINT"Giro No"F
490 IF F> 10 THEN PRINT:PRINT"Tu giro cubista nu
mero"F-10 :PRINT
500 IF F< 11 THEN RETURN
510 P=0
520 IF B(1)=1 AND B(2)=1 AND B(3)=2 AND B(4)=2 A
ND B(5)=1 AND B(6)=1 AND B(7)=2 AND B(8)=2 THEN P=
P+1
530 IF B(9)=3 AND B(10)=3 AND B(11)=4 AND B(12)=
4 AND B(13)=3 AND B(14)=3 AND B(15)=4 AND B(16)=4
THEN P=P+1
540 IF P<>2 THEN RETURN
550 PRINT:PRINT"Lo has resuelto en solo"F-10"gir
os"
560 PRINT:PRINT"Bien hecho, cubista!"
```

MUMBLE MARBLE

Nuestro siguiente programa es una versión para ordenador de un juego que quizá el lector conozca con el nombre de «Solitaire», en el cual el jugador tiene que mover unas canicas en un tablero. Al principio del juego sólo está vacío el agujero central. El jugador puede saltar sobre una canica para ocupar los agujeros vacíos. La idea del juego es terminar con una sola canica en el agujero central.

Se han contado muchas historias sobre los orígenes de este juego. La más interesante de estas historias es la que dice que fue ideado por un prisionero de la Bastilla, quien intentaba con ello aliviar la monotonía de su celda. Independientemente de que esta historia sea cierta, de lo que no cabe duda es de que el juego crea una adicción, puesto que el jugador no parará hasta encontrar el medio infalible de resolver el problema que plantea.

Este es el aspecto de la pantalla al principio del juego:

```
Primero, la coordenada lateral
     (99 para rendirte)

     1 2 3 4 5 6 7
         0 0 0      1
         0 0 0      2
     0 0 0 0 0 0 0  3
     0 0 0 * 0 0 0  4     Hasta ahora ha habido 0 movimientos
     0 0 0 0 0 0 0  5
         0 0 0      6
         0 0 0      7     Que piedra quieres mover?
```

```
Primero, la coordenada lateral
     (99 para rendirte)


     1 2 3 4 5 6 7
         0 0 0      1
         0 * 0      2
     0 * 0 0 0 0 0  3    Hasta ahora ha habido 4 movimientos
     0 0 * 0 0 0 0  4
     0 0 0 0 0 0 0  5
         * 0 0      6    Que piedra quieres mover?41
         * 0 0      7    41 a donde?43
```

Este es el listado:

```
 10 REM Mumble Marble
 20 GOSUB 410
 30 GOSUB 250
 40 REM ACEPTA EL MOVIMIENTO
 50 PRINT"Que piedra quieres mover";
 60 INPUT A
 70 IF A=99 THEN GOTO 240
 80 IF A<11 OR A>77 THEN GOTO 50
 90 IF A(A)<>79 THEN GOTO 50
100 PRINTA"a donde";
110 INPUT B
120 IF B<11 OR B> 77 THEN GOTO 110
130 IF A(B)<>E THEN GOTO110
140 A((A+B)/2)=E:A(A)=E:A(B)=79
150 MOVI=MOVI+1
160 CUENTA=0
170 FOR F=11 TO 75
180 IF A(F)=79 THEN CUENTA=CUENTA+1
190 NEXT F
200 GOSUB 250
210 PRINT"Hay"CUENTA"piedras en el tablero"
220 IF CUENTA<>1 THEN 40
```

```
230 IF A(44)=79 THEN PRINT"Lo, conseguistes en s
olo"MOVI"movimientos!":END
240 PRINT"El juego ha terminado y no lo has cons
eguido!":END
250 REM Impresion
260 CLS:REM Pon HOME aqui si tu ordenador lo tie
ne.
270 PRINT"Primero, la coordenada lateral"
280 PRINTTAB(5);"(99 para rendirte)"
290 PRINT:PRINT
300 PRINT"     1 2 3 4 5 6 7"
310 PRINT TAB (5);
320 FOR D=11 TO 75
330 T=10*(INT(D/10))
340 IF D-T=8 THEN D=D+2:PRINT T/10:PRINT TAB(5);
:GOTO 360
350 PRINT CHR$(A(D));" ";
360 NEXT D:PRINT"     7"
370 PRINT:PRINT:PRINT
380 PRINT"Hasta ahora ha habido"MOVI"movimientos
"
390 PRINT:PRINT
400 RETURN
410 REM Inicializacion
420 CLS
430 DIM A(87)
440 E=42
450 FOR D= 11 TO 75
460 T=10*(INT(D/10))
470 IF D-T = 8 THEN D=D+3
480 READA(D)
490 NEXT D
500 MOVI=0
510 RETURN
520 REM 42 es el ASC("*")
530 REM 79 es el ASC("O")
540 DATA 32,32,79,79,79,32,32
550 DATA 32,32,79,79,79,32,32
560 DATA 79,79,79,79,79,79,79
570 DATA 79,79,79,42,79,79,79
580 DATA 79,79,79,79,79,79,79
590 DATA 32,32,79,79,79,32,32
600 DATA 32,32,79,79,79
```

CICLOTRON

En CICLOTRON el jugador hace entrar los átomos por la parte superior de la máquina especificando para ello un número y posteriormente intenta localizar las partículas de antimateria escondidas en la máquina a partir del comportamiento del átomo.

Si el átomo golpea una partícula de antimateria, puede ser absorbido por ésta o ser desviado hacia la derecha o hacia la izquierda. Un átomo puede verse afectado por más de una partícula de antimateria lo cual hace la resolución del problema aún más difícil.

Siempre hay tres partículas de antimateria en la máquina, aunque todas pueden muy bien encontrarse en el mismo número ciclotron. El jugador cuenta con un número muy limitado de bazas o jugadas para encontrar la antimateria. En cada jugada el jugador puede escoger entre introducir el número en donde quiera meter un átomo o bien ser un poco imprudente y pulsar 8, lo que le da la posibilidad de adivinar la posición de las partículas de antimateria.

Este es el aspecto del programa cuando se ejecuta:

```
Tienes 6 segundos
Dime el numero del ciclotron
o 8 si crees que sabes
donde esta la antimateria
?3
```

```
        3
- - - - - - -
- - - - - - -
- - - - - - -
- - - - - - -
- - - - - - -
        3
El atomo salio de 3
```

```
Tienes 1 segundos
Dime el numero del ciclotron
o 8 si crees que sabes
donde esta la antimateria
?8
Donde crees que esta
la antimateria?
?1,6,5
Localizastes 1 correctamente


Lo siento, se ha acabado el tiempo

La antimateria estaba en
 7  1  7
```

Y este es el listado

```
 10 REM Ciclotron
 20 CLS
 30 N=0:MT=0:DIM B(7),Q(3),W(3)
 40 PRINT"Aprieta una tecla cuando"
 50 PRINT"estes preparado para empezar"
 60 N=N+1:IF INKEY$="" THEN 60
 70 RANDOMIZE
 80 REM Situa antimateria
 90 FOR A=1 TO 3
100 B(A)=INT(RND(-1)*7)+1
110 NEXT A
120 FOR H=6 TO 1 STEP-1
130 CLS
140 PRINT:PRINT:PRINT"Tienes"H"segundos"
150 PRINT"Dime el numero del ciclotron"
160 PRINT"o 8 si crees que sabes"
170 PRINT"donde esta la antimateria"
180 INPUT C
```

```
190 IF C=8 THEN 520
200 GOSUB 450
210 D=1
220 IF B(D)=C THEN 350
230 IF D<3 THEN D=D+1:GOTO 220
240 GOSUB 500
250 PRINT"El atomo salio de"C
260 FOR Z=1 TO 2000:NEXT Z
270 NEXT H
280 PRINT:PRINT:PRINT
290 PRINT"Lo siento, se ha acabado el tiempo"
300 PRINT:PRINT"La antimateria estaba en"
310 FOR A=1 TO 3
320 PRINT B(A);
330 NEXT A
340 END
350 REM Actuacion de la antimateria
360 IF RND(-N)>.9 THEN 420
370 IF C=1 THEN C=2:GOTO 210
380 IF C=7 THEN C=6  :GOTO 210
390 F=INT(RND(-N)*2)-1
400 C=C+F
410 GOTO 210
420 REM La antimateria absorbe un atomo
430 PRINT"El atomo ha sido absorbido"
440 GOTO 260
450 PRINT:PRINT:PRINTTAB (3+2*C);C
460 FOR Z=1 TO 5
470 PRINT TAB(6);"- - - - - - -"
480 NEXT Z
490 RETURN
500 PRINT TAB(3+2*C);C
510 RETURN
520 REM Intento de localizacion
530 PRINT"Donde crees que esta"
540 PRINT"la antimateria?"
550 INPUTQ(1),Q(2),Q(3)
560 FOR T=1 TO 3:W(T)=B(T):NEXT T
570 SR=0
580 FOR X=1 TO3:FOR Y=1 TO 3
590 IF Q(X)=B(Y) THEN SR=SR+1:B(Y)=0:Q(X)=0
600 NEXT Y:NEXT X
610 GOTO650
```

```
620 FOR P=1 TO 3
630 B(P)=W(P)
640 NEXT P
650 IF SR>2 THEN 700
660 IF SR>0 THEN PRINT"Localizastes"SR"correctam
ente"
670 IF SR=0 THEN PRINT"Estas completamente equiv
ocado!"
680 FOR T=1 TO 3:B(T)=W(T):NEXT T
690 GOTO260
700 PRINT"Bien hecho. Localizastes la"
710 PRINT"antimateria con"H"segundos"
720 PRINT"de sobra..."
730 IF 6-H<MT THEN MT=6-H
740 PRINT:PRINT"Tu mejor tiempo hasta ahora son"
MT"segundos"
750 FOR Z=1 TO 2000:NEXT Z
760 GOTO90
```

SWITCHEROO

En este juego se le presentan al jugador los números del 1 al 9 colocados en un orden aleatorio y éste tiene que conseguir ponerlos en orden consecutivo del 1 al 9 en el menor número de jugadas posible.

El jugador introduce sus jugadas en forma de números y el ordenador lleva a cabo un «switcheroo» utilizando el número que el jugador ha introducido. Funciona así:

```
Un momento, por favor...
9

95                          Movimiento numero 1

956

9562                        956283174

95628                       Que numero conmutamos?

956283

9562831

95628317

956283174
```

```
Movimiento numero 73

123456798

Que numero conmutamos?

123456789

Lo has hecho, campeon!

Solo has necesitado 73 movimientos...
```

Este es el listado del programa:

```
 10 REM Switcheroo
 20 CLS
 30 PRINT:PRINT"Un momento, por favor..."
 40 GOSUB 100
 50 GOSUB 230
 60 IF A$="123456789" THEN 400
 70 M=M+1
 80 GOTO50
 90 END
100 RANDOMIZE
110 M=1:X=0
120 A$=""
130 FOR T=1 TO 9
140 L=INT(RND(-1)*9)+49
150 Q=1
160 IF MID$(A$,Q,1)=CHR$(L) THEN 140
```

```
170 IF Q<T THEN Q=Q+1:GOTO 160
180 A$=A$+CHR$(L)
190 PRINTA$
200 PRINT
210 NEXTT
220 RETURN
230 REM Impresion
240 CLS
250 PRINT:PRINT:PRINT
260 PRINT"Movimiento numero"M
270 PRINT:PRINT:PRINT
280 PRINTA$
290 PRINT:PRINT"Que numero conmutamos?"
300 IF INKEY$<>"" THEN 300
310 C$=INKEY$
320 R=VAL(C$)
330 IF R<1 OR R>8 THEN 310
340 B$=""
350 FOR T=9 TO R STEP-1
360 B$=B$+MID$(A$,T,1)
370 NEXT T
380 A$=LEFT$(A$,R-1) + B$
390 RETURN
400 PRINT:PRINT:PRINT
410 PRINTA$
420 PRINT:PRINT:PRINT
430 PRINT"Lo has hecho, campeon!"
440 PRINT:PRINT"Solo has`necesitado"M"movimiento
s..."
450 END
```

FLIPPER

Este intrigante juego puede proporcionar al jugador un grado considerable de apertura y agilidad mental. Cuando ejecuta el programa el jugador verá una mezcla de X y * en una red de tres por tres. Tendrá que terminar la partida con una X en el centro y ocho * rodeándola.

En primer lugar el jugador introduce el número del lugar que quiere golpear con el «flipper». Cuando se golpea una pieza situada en una esquina, las piezas que están junto a ésta cambian y se convierten en sus opuestas (es decir, X se convierte en * y a la inversa). Si se golpea la pieza central por un lado cambian las dos que están situadas a sus lados y golpear la pieza del centro cambia las piezas de cada lado.

La pieza que es golpeada cambia siempre en cada caso.

Este es el aspecto que tiene en acción:

```
1 2 3      * * *

4 5 6      * * *

7 8 9      * * X

El numero de'X' es 1

Solo necesitas una - en el cuadrado central

Cual quieres mover?
```

```
Has movido el 8

1  2  3        * * X

4  5  6        * X *

7  8  9        * X *

Ese fue el movimiento No 4

El numero de'X' es 3

Solo necesitas una - en el cuadrado central

Cual quieres mover?



Has movido el 5

1  2  3        * * *

4  5  6        * X *

7  8  9        * * *


Lo has resuelto en solo 22 movimientos
```

Y este es el listado:

```
 10 REM Flipper
 20 DIM A(10),F(4)
 30 M=-1:Q=42:X=88:P=0
 40 RANDOMIZE
 50 FOR C=1 TO 9
 60 A(C)=Q
 70 IF INT(RND(-1)+.5)=0 THEN A(C)=X
 80 NEXT C
 90 GOSUB 270
100 M=M+1
110 N=0
120 FOR C=1 TO 9
130 IF A(C)=X THEN N=N+1
140 NEXT C
150 IF N=1 AND A(5) = X THEN 350
160 IF M> 0 THEN PRINT :PRINT"Ese fue el movimie
nto No"M
170 PRINT:PRINT"El numero de'X' es"N
180 PRINT:PRINT"Solo necesitas una - en el cuadr
ado central"
190 PRINT:PRINT"Cual quieres mover?"
200 IF INKEY$<>"" THEN 200
210 A$= INKEY$
220 N=VAL(A$):IF N<1 OR N>9 THEN 210
230 P=N
240 GOSUB 380
250 GOTO90
260 END
270 CLS:PRINT:PRINT
280 IF P<>0 THEN PRINT"Has movido el"P
290 PRINT:PRINT:PRINT"1  2  3",CHR$(A(1));" ";CH
R$(A(2));" ";CHR$(A(3))
300 PRINT
310 PRINT"4  5  6",CHR$(A(4));" ";CHR$(A(5));" "
;CHR$(A(6))
320 PRINT
330 PRINT"7  8  9",CHR$(A(7));" ";CHR$(A(8));" "
;CHR$(A(9))
340 RETURN
```

```
350 PRINT:PRINT
360 PRINT"Lo has resuelto en solo"M"movimientos"
370 END
380 IF A(N)=X THEN RETURN
390 IF N=1 THEN F(1)=2:F(2)=4:F(3)=5:F(4)=10
400 IF N=2 THEN F(1)=1:F(2)=3:F(3)=10:F(4)=10
410 IF N=3 THEN F(1)=2:F(2)=5:F(3)=6 :F(4)=10
420 IF N=4 THEN F(1)=1:F(2)=7:F(3)=10:F(4)=10
430 IF N=5 THEN F(1)=2:F(2)=4:F(3)=8 :F(4)=6
440 IF N=6 THEN F(1)=3:F(2)=9:F(3)=10:F(4)=10
450 IF N=7 THEN F(1)=4:F(2)=5:F(3)=8 :F(4)=10
460 IF N=8 THEN F(1)=7:F(2)=9:F(3)=10:F(4)=10
470 IF N=9 THEN F(1)=8:F(2)=5:F(3)=6 :F(4)=10
480 FOR G= 1 TO 4
490 F=0
500 IF A(F(G)) = X THEN F=1
510 IF F=1 THEN A(F(G)) = Q
520 IF F=0 AND A(F(G)) = Q THEN A(F(G))=X
530 NEXT G
540 A(N)=X
550 RETURN
```

EN BUSCA DE DARYL

Para terminar tenemos el juego más problemático de todos. Daryl se pierde en un sistema de cuevas subterráneo de una gran complejidad.

El jugador tiene que entrar en el sistema, abrirse camino para encontrar a Daryl que se encuentra en la cueva 29 y después volver a salir.

Las cavernas están conectadas por un conjunto de túneles coherente y estable y lo que tiene que hacer el jugador es intentar solucionar el enigma que supone llegar hasta Daryl, para lo cual ha de ir haciendo un mapa a medida que descubre el modo en que están conectadas las diferentes cuevas. El jugador ha de saber que hay túneles que pasan por debajo de otros. Las cuevas no están dispuestas en un claro orden numérico. Más bien, los números de las cuevas no son sino meros puntos de referencia para el jugador.

Es posible que el jugador encuentre que el problema tiene una difícil solución, pero en realidad no es en modo alguno imposible. Se deben examinar con cuidado las referencias relativas a las direcciones (como N para el norte y sucesivamente). De hecho, el norte no está arriba de la cueva ni el sur abajo. Más bien, se refieren a direcciones como si todo el sistema se observara a vista de pájaro y la dirección en la parte superior del sistema fuera el norte, el este la derecha y así sucesivamente. En cualquier caso, lo más probable es que las claves de direcciones le confundan la primera vez que ejecute el programa.

No abandone así como así. Puede lograrse. El número de caminos posibles es extraordinariamente alto, así que una vez que lo ha logrado de un modo, es posible que quiera intentarlo de otro. A los veteranos de este programa les suele gustar encontrar a Daryl siguiendo un camino diferente cada vez que lo ejecutan y después de haberlo encontrado vuelven a la entrada (llevándole) por un camino diferente del anterior.

Como prueba de que no es un problema sin solución, lo que se ofrece a continuación es un camino hasta Daryl seguido de otro para salir del sistema:

```
Estas en camino para encontrar a
Daryl, atrapado en este sistema de cuevas

Estas en la cueva No 1

Hay tuneles hacia:
 2  7  10

Daryl esta hacia el norte de ti


Tu Daryl-proximetro mide 30


A que cueva quieres ir?2
```

```
Estas en camino para encontrar a
Daryl, atrapado en este sistema de cuevas

Estas en la cueva No 2

Hay tuneles hacia:
 25  11  1  8

Daryl esta hacia el norteeste de ti


Tu Daryl-proximetro mide 12


A que cueva quieres ir?8
```

```
Estas en camino para encontrar a
Daryl, atrapado en este sistema de cuevas

Estas en la cueva No 8

Hay tuneles hacia:
 26  2

Daryl esta hacia el norteeste de ti


Tu Daryl-proximetro mide 33


A que cueva quieres ir?26
```

```
Estas en camino para encontrar a
Daryl, atrapado en este sistema de cuevas

Estas en la cueva No 26

Hay tuneles hacia:
 8  3  23  12

Daryl esta hacia el norteeste de ti


Tu Daryl-proximetro mide 54


A que cueva quieres ir?23
```

```
Estas en camino para encontrar a
Daryl, atrapado en este sistema de cuevas

Estas en la cueva No 23

Hay tuneles hacia:
 18  26

Daryl esta hacia el norteeste de ti


Tu Daryl-proximetro mide 86


A que cueva quieres ir?18



Estas en camino para encontrar a
Daryl, atrapado en este sistema de cuevas

Estas en la cueva No 18

Hay tuneles hacia:
 13  15  22  23

Daryl esta hacia el norteoeste de ti


Tu Daryl-proximetro mide 54


A que cueva quieres ir?22
```

Estas en camino para encontrar a
Daryl, atrapado en este sistema de cuevas

Estas en la cueva No 22

Hay tuneles hacia:
 18 29

Daryl esta hacia el norteoeste de ti

Tu Daryl-proximetro mide 74

A que cueva quieres ir?29

Bien hecho.Has encontrado a Daryl
aparentemente a salvo y bien ....

Has encontrado a Daryl, y
vuelves con el a la entrada

Estas en la cueva No 29

Hay tuneles hacia:
 22

La salida esta hacia el norteoeste de ti

Tu Salida-proximetro mide 55

A que cueva quieres ir?22

```
Has encontrado a Daryl, y
vuelves con el a la entrada

Estas en la cueva No 22

Hay tuneles hacia:
 18  29

La salida esta hacia el norteoeste de ti


Tu Salida-proximetro mide 44


A que cueva quieres ir?18
```

```
Has encontrado a Daryl, y
vuelves con el a la entrada

Estas en la cueva No 18

Hay tuneles hacia:
 13  15  22  23

La salida esta hacia el norteoeste de ti


Tu Salida-proximetro mide 24


A que cueva quieres ir?15
```

```
Has encontrado a Daryl, y
vuelves con el a la entrada

Estas en la cueva No 15

Hay tuneles hacia:
 18  6

La salida esta hacia el suroeste de ti

Tu Salida-proximetro mide 15

A que cueva quieres ir?6
```

```
Has encontrado a Daryl, y
vuelves con el a la entrada

Estas en la cueva No 6

Hay tuneles hacia:
 15  19

La salida esta hacia el este de ti

Tu Salida-proximetro mide 5

A que cueva quieres ir?19
```

```
Has encontrado a Daryl, y
vuelves con el a la entrada

Estas en la cueva No 19

Hay tuneles hacia:
 20  6

La salida esta hacia el sureste de ti


Tu Salida-proximetro mide 46


A que cueva quieres ir?20
```

```
Has encontrado a Daryl, y
vuelves con el a la entrada

Estas en la cueva No 20

Hay tuneles hacia:
 19  14  21

La salida esta hacia el sureste de ti


Tu Salida-proximetro mide 25


A que cueva quieres ir?21
```

```
Has encontrado a Daryl, y
vuelves con el a la entrada

Estas en la cueva No 21

Hay tuneles hacia:
 11  25  20

La salida esta hacia el sureste de ti

Tu Salida-proximetro mide 63

A que cueva quieres ir?25
```

```
Has encontrado a Daryl, y
vuelves con el a la entrada

Estas en la cueva No 25

Hay tuneles hacia:
 21  2

La salida esta hacia el sureste de ti

Tu Salida-proximetro mide 43

A que cueva quieres ir?2
```

Has encontrado a Daryl, y
vuelves con el a la entrada

Estas en la cueva No 2

Hay tuneles hacia:
 25 11 1 8

La salida esta hacia el sureste de t

Tu Salida-proximetro mide 22

A que cueva quieres ir?1

Eres un heroe!!!

Has conseguido regresar con
Daryl sano y salvo...

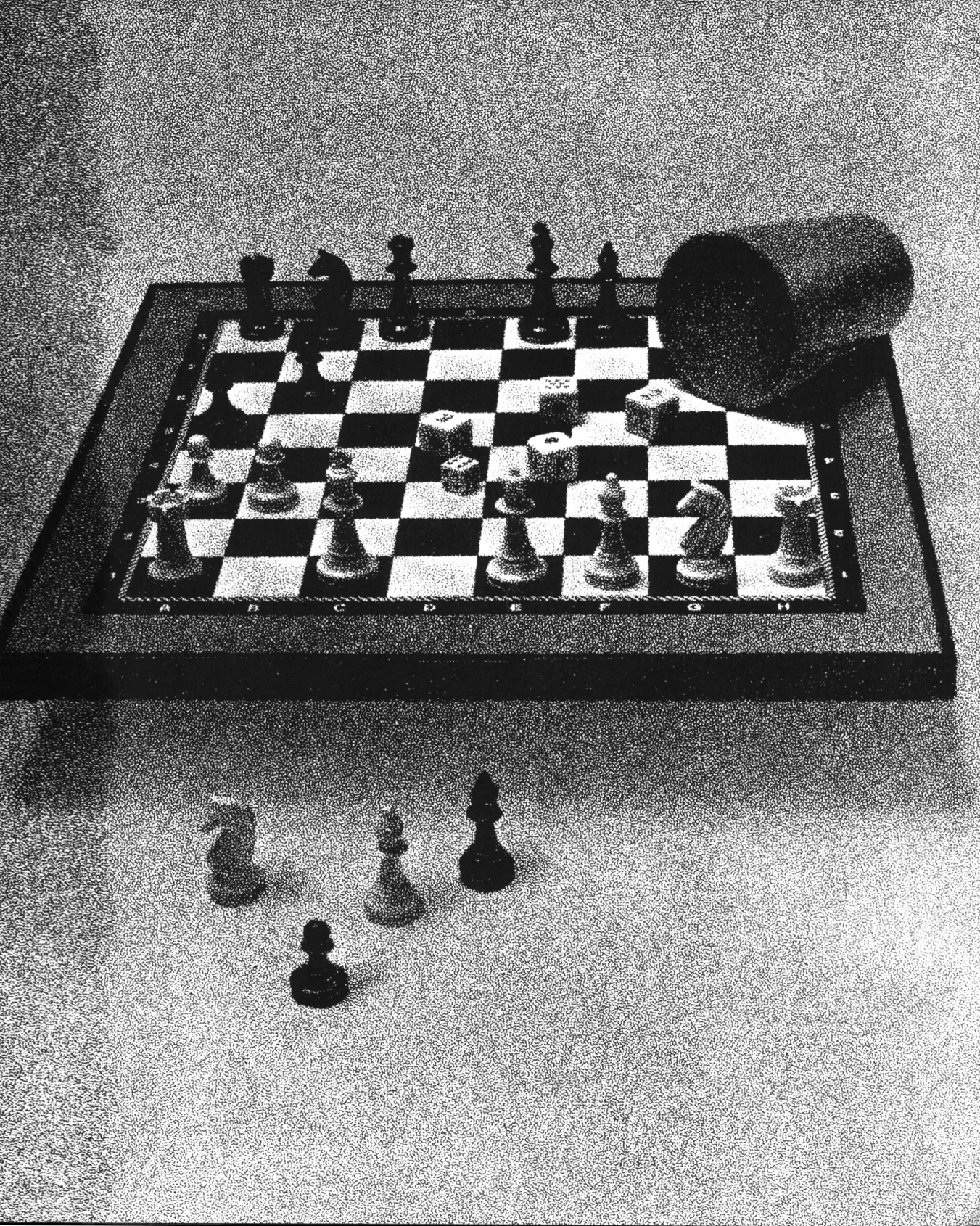
Ve derecho a la Moncloa
para ser condecorado por el
Presidente.....
Nos volveremos a ver, heroe!

Y este es el listado:

```
   10 REM En busca de Daryl
   20 GOSUB 770:REM Inicializacion
   30 GOSUB 480:REM Descripcion del lugar
   40 GOSUB 360:REM Movimiento
   50 IF DA=0 AND CUEVA=29 THEN GOSUB 90:REM Has e
ncontrado a Daryl
   60 IF DA=1 AND CUEVA=1 THEN 210:REM Vuelta a ca
sa
   70 GOTO30
   80 REM ************************
   90 REM    Daryl encontrado
  100 REM ************************
  110 PRINT:PRINT"Bien hecho.Has encontrado a Dary
l"
  120 PRINT"aparentemente a salvo y bien ...."
  130 FOR P=1 TO 1000:NEXT P
  140 PRINT:PRINT"Es hora de intentar"
  150 PRINT"volver a la salida, con"
  160 PRINT"Daryl..."
  170 FOR P= 1 TO 1000:NEXT P
  180 DA=1
  190 RETURN
  200 REM ************************
  210 REM    Vuelta a casa
  220 REM ************************
  230 PRINT:PRINT
  240 PRINT"Eres un heroe!!!"
  250 FOR P=1 TO 1000:NEXT P
  260 PRINT:PRINT"Has conseguido regresar con"
  270 PRINT"Daryl sano y salvo..."
  280 PRINT:PRINT
  290 PRINT"Ve derecho a la Moncloa"
  300 PRINT"para ser condecorado por el"
  310 PRINT"Presidente....."
  320 FOR P=1 TO 1000:NEXT P
  330 PRINT"Nos volveremos a ver, heroe!"
  340 END
  350 REM ******************************
  360 REM            Movimiento
  370 REM ******************************
```

```
380 PRINT:PRINT
390 INPUT"A que cueva quieres ir";J
400 BANDERA=0
410 FOR K=1 TO 4
420 IF Z(CUEVA,K)=J THEN BANDERA=1
430 NEXT K
440 IF BANDERA=0 THEN PRINT"No hay tunel a la cu
eva"J:FOR P=1 TO 1000:NEXT P:GOTO390
450 CUEVA=J
460 RETURN
470 REM ******************************
480 REM   Descripcion de los lugares
490 REM ******************************
500 CLS:PRINT:PRINT
510 IF DA=0 THEN PRINT"Estas en camino para enco
ntrar a":PRINT"Daryl, atrapado en este sistema de
cuevas"
520 IF DA=1 THEN PRINT"Has encontrado a Daryl, y
":PRINT"vuelves con el a la entrada"
530 PRINT:PRINT"Estas en la cueva No"CUEVA
540 PRINT:PRINT"Hay tuneles hacia:"
550 FOR Q=1 TO 4
560 IF Z(CUEVA,Q)>0 THEN PRINTZ(CUEVA,Q);
570 NEXT Q
580 PRINT:PRINT
590 IF DA=1 THEN 660
600 PRINT"Daryl esta hacia el ";
610 IF Z(CUEVA,5)>4 THEN PRINT"norte";
620 IF Z(CUEVA,5)<4 THEN PRINT"sur";
630 IF Z(CUEVA,6)<7 THEN PRINT"este";
640 IF Z(CUEVA,6)>7 THEN PRINT"oeste";
650 GOTO710
660 PRINT"La salida esta hacia el ";
670 IF Z(CUEVA,5)>7 THEN PRINT"norte";
680 IF Z(CUEVA,5)<7 THEN PRINT"sur";
690 IF Z(CUEVA,6)<7 THEN PRINT"este";
700 IF Z(CUEVA,6)>7 THEN PRINT"oeste";
710 PRINT" de ti"
720 PRINT:PRINT
730 IF DA=0 THEN PRINT"Tu Daryl-proximetro mide"
10*(ABS(Z(CUEVA,5)-4))+ABS(Z(CUEVA,6)-7)
740 IF DA=1 THEN PRINT"Tu Salida-proximetro mide
"10*(ABS(Z(CUEVA,5)-7))+ABS(Z(CUEVA,6)-7)
```

```
 750 RETURN
 760 REM *********************
 770 REM     Inicializacion
 780 REM *********************
 790 CLS
 800 DIM Z(29,6)
 810 FOR H=1 TO 29
 820 READB:READC:READD
 830 READE:READF:READG
 840 Z(H,1)=B:Z(H,2)=C:Z(H,3)=D
 850 Z(H,4)=E:Z(H,5)=F:Z(H,6)=G
 860 NEXT H
 870 DA=0:REM Aun no has encontrado a Daryl
 880 CUEVA = 1:REM Posicion inicial
 890 RETURN
 900 DATA 2,7,10,0,7,7,25,11,1,8,5,5
 910 DATA 26,0,0,0,10,1,24,9,7,0,2,10
 920 DATA 27,28,0,0,10,8,15,19,0,0,7,2
 930 DATA 4,1,0,0,5,8,26,2,0,0,7,4
 940 DATA 4,24,10,0,4,11,1,9,0,0,7,9
 950 DATA 21,2,0,0,2,7,26,28,0,0,11,4
 960 DATA 18,16,0,0,8,10,20,0,0,0,2,2
 970 DATA 18,6,0,0,6,12,13,17,0,0,9,5
 980 DATA 16,28,0,0,11,6,13,15,22,23,9,11
 990 DATA 20,6,0,0,3,1,19,14,21,0,5,2
1000 DATA 11,25,20,0,1,4,18,29,0,0,11,11
1010 DATA 18,26,0,0,12,1,4,9,0,0,1,12
1020 DATA 21,2,0,0,3,4,8,3,23,12,9,3
1030 DATA 5,0,0,0,9,7,17,5,12,0,12,7
1040 DATA 22,0,0,0,12,12
```


A B C D E F G H

PARA CREAR JUEGOS PROPIOS

Puede suceder que cuando lleve un tiempo programando, encuentre que, aunque domina bien muchas de las técnicas necesarias para escribir sus propios juegos, le faltan ideas. Si es este su caso, le interesará este capítulo. Intentaré esbozar unos cuantos juegos que me parecen ideales para convertir en juegos de ordenador.

En este momento preparo un nuevo libro de juegos y estoy buscando nuevos programas para incluir en ese volumen. Si tú, lector, has escrito algunos juegos que pienses que pueden ser adecuados (es decir, que merecen la pena por ser originales y no dependientes de la máquina), me gustaría mucho verlos con el fin de comprarlos para utilizarlos en el próximo volumen. Por supuesto, tu nombre aparecerá publicado junto con el programa. Sólo tomaré en consideración los listados impresos, salidos directamente del ordenador. Puedes escribirme enviando tu carta a mi editor.

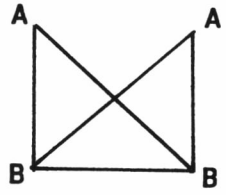
La razón de mencionar esto es que me encantaría considerar nuevos programas basados en las ideas que daré en este capítulo o en otras diferentes.

Volvamos ahora al punto en el que empezamos. A no ser que se esté informatizando un juego muy conocido, como las Damas, por ejemplo, en cuyo caso se espera que el programa coincida en todos los aspectos con la versión no informatizada, no es necesario que la versión para ordenador de un determinado juego siga al pie de la letra las reglas del juego que yo bosquejo a continuación. Lo más probable es que, después de cierto grado de desarrollo, el juego se despega de la idea inicial y puede terminar pareciéndose muy poco a su «progenitor». Todo sea para bien.

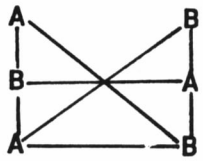
JUEGOS PARA ADAPTAR

SHUFFLEBOARD.—Los jugadores deslizan unas fichas planas y circulares en un tablero dirigiéndolas hacia un «blanco» triangular que contiene círculos pintados con números dentro de ellos. La ficha que termine enteramente en un círculo numerado proporciona al jugador los puntos que contenga ese número. Este juego tiene muchas posibles variaciones entre las que se incluyen los juegos del tipo de los dardos. Una vez que consiga ejecutar un programa de este tipo de juegos, puede hacerle los cambios superficiales precisos para convertirlo en otro juego.

HORSESHOE.—Empezando con un tablero como éste, con las fichas de un jugador situadas en las posiciones marcadas con A y las del otro en las que están marcadas con B, los jugadores mueven por turno siguiendo una de las líneas para colocarse en un punto vacío. El juego termina cuando uno de los jugadores descubre que no puede mover. A pesar de su sencillez, es un juego muy interesante, como se puede ver al jugarlo.



DAMOCLES.—Este juego se parece al anterior, pero usa un tablero más grande. El objetivo del juego es también el mismo que el del anterior y usa este tablero:



MU-TORERE.—Este juego, cuyo origen se encuentra entre los maorís de Nueva Zelanda, se juega en un tablero que tiene la forma de una estrella de ocho puntas en la que cada punta está unida a un círculo en el medio. Este círculo se llama *putahi.* Los jugadores empiezan con cuatro fichas que sitúan en la parte exterior de cuatro puntas consecutivas. El objetivo del juego, como en los dos anteriores, es conseguir que el otro jugador no pueda mover. Cada jugador puede mover de un punto al punto siguiente siempre que éste esté vacío o desde el putahi a cualquier otro punto vacío. Otra posibilidad es mover desde cualquier punto al putahi, pero esto sólo puede hacerse cuando al menos uno de los puntos que están al lado del punto desde el que mueve el jugador esté ocupado por una ficha del contrincante.

BINGO.—Este juego se juega normalmente con cartones marcados con números que se van tachando según se van sacando los mismos números de un bombo siendo el ganador el primero que consiga tachar una línea entera de números en cualquier dirección. Este juego se puede simplificar mucho para su uso en el ordenador; éste puede introducir dos cartones de modo aleatorio utilizando números entre el cero y el 99 e imprimiéndolos en una cuadrícula de cinco por cinco. Posteriormente el ordenador puede escoger números de modo aleatorio entre el cero y el 99, tachar los números en los cartones y vigilar si ha ganado alguno de los dos contrincantes. Este programa puede hacerse todavía más interesante si, en vez de generar sencillamente un número de modo aleatorio, se incluyen dos esferas con flechas giratorias. La primera de las esferas, las «decenas», está marcada de cero a nueve y se hace girar primero. La segunda esfera está marcada como la primera, pero representa las «unidades».

JUEGOS DE DADOS.—Como ya indicaba en la introducción del capítulo dedicado a estos juegos, los juegos de dados son relativamente fáciles de convertir en programas para jugar con el ordenador, especialmente si no se espera que el ordenador demuestre una inteligencia sorprendente al jugar. Hay bastantes juegos de dados que se pueden intentar adaptar al ordenador. Entre estos se incluyen:

DROP DEAD.—Se necesitan cinco dados para jugar a este juego. El primer jugador lanza los dados y todos los que caigan en dos o en cinco se retiran. El total del resto se escribe y los dados que cayeron en dos o en cinco se vuelven a tirar. Después de esta tirada, se vuelven a retirar los dados que hayan caído de nuevo en dos o en cinco y el total del resto de los dados se suma al total conseguido en la primera tirada. Este proceso continúa hasta que no quedan dados, terminando en ese momento la baza de ese jugador e iniciándose la del segundo quien repite el mismo procedimiento. El ganador es el que haya conseguido más puntos. Se puede jugar una partida a tres juegos o manos o limitar el número total de tiradas que puede hacer por baza cada jugador, independientemente de que le queden dados o no.

MARTINETTI.—Para este juego se necesitan tres dados, una hoja que haga de marcador con una fila de casillas numeradas de uno a doce y dos fichas, una para cada jugador. El objetivo del juego es atravesar los números por orden, antes de que lo haga el contrario. Los jugadores tiran los dados por turno; si uno de los dados cae en uno, la ficha de ese jugador se pone en la casilla número uno, si también ha caído otro dado en dos, la ficha puede pasarse a la número dos. Después de la casilla uno, los jugadores pueden escoger entre utilizar los puntos de un solo dado para moverse a la siguiente casilla o sumar el resultado de todos para ver si sale el número que necesitan para avanzar a la siguiente casilla. Hay una variación de este juego (que se suele llamar EVEREST) que consiste en descender una vez que se ha llegado a doce. El ganador de EVEREST es el jugador que consiga llegar primero al campamento base, es decir, volver a la casilla número uno.

DADOS INDIOS.—Este juego es similar al POKER DE DADOS y se juega con cinco dados. El objetivo es sencillamente conseguir las mejores combinaciones de dados. No se pueden conseguir,

como en el póker de cartas, cartas adicionales tras el descarte. Se tiran los dados una sola vez y esa es la mano del jugador. La clasificación de las manos es la siguiente:

— *Cinco del mismo tipo (cinco sotas, por ejemplo).*
— *Cuatro del mismo tipo.*
— *Full (tres del mismo tipo y una pareja).*
— *Dos parejas.*
— *Una pareja.*
— *El dado más alto.*

BIDOU.— Este juego también es similar al POKER DE DADOS, sin embargo, el objetivo del jugador es deshacerse de las fichas más que acumularlas. Gana la partida el jugador que ha perdido más juegos o manos. Este juego se juega con tres dados y una cantidad determinada de fichas que se distribuyen por igual entre los dos jugadores. Se hace una apuesta y se iguala antes de que los jugadores tiren los dados por turno. Las jugadas se clasifican como sigue:

2 1 1; 2 2 1; 4 2 1; 6 6 6; 5 5 5; 4 4 4;

3 3 3; 2 2 2; 1 1 1; 3 3 6; 3 3 5; 3 3 4;

3 3 2; 3 3 1; 1 1 6; 1 1 5; 1 1 4; 1 1 3;

3 2 1; 4 3 2; 5 4 3; 6 5 4

Si la combinación que sale tras tirar los dados no aparece en esta lista, se suman los puntos y gana el que haya conseguido más (que en realidad, en este juego, pierde).

BARBOOTH.—También se le conoce con el nombre de BARBUDI y se juega con dos dados. Se sugiere aquí que los jugadores empiecen la partida con una banca de 30 fichas que se repartirán a partes iguales entre los dos jugadores, 15 para cada uno, siendo este el capital inicial de que disponen para las apuestas y cobrándoles dos fichas por cada jugada. Los jugadores tiran los dados por turno. Ganan cuando les sale 3 3; 5 5; 6 5; y 6 6. Pierden si les sale 1 1; 2 2;

2 1 y 4 4. Si los dos consiguen una de las combinaciones ganadoras, gana el que haya conseguido la combinación más próxima a la primera de la lista (es decir, 3 3 es mejor que 6 6). Si los dos tienen la misma combinación ganadora, el dinero va a la banca. Si uno de los jugadores consigue una combinación que no es ni ganadora ni perdedora y al otro le sale una combinación perdedora, las fichas del perdedor van a la banca y las del otro vuelven a su propietario.

Si a uno de los jugadores le sale una combinación ganadora y al otro una que no es ni ganadora ni perdedora, el ganador recupera sus fichas así como las apostadas por el otro. Sin embargo, si un jugador consigue una combinación ganadora y el otro una perdedora, el banco contribuye con el mismo número de fichas que se han apostado en total. El juego continúa hasta que alguno de los jugadores o la banca se han quedado sin fichas. El jugador que tenga más fichas en ese momento es el ganador. Observe que este modo de distribuir las ganancias y las pérdidas no es el mismo que cuando BARBOOTH se juega de modo «profesional», pero es un medio de pagar a los ganadores relativamente fácil de informatizar y contribuye a que el juego sea más divertido.

FIVE THOUSAND.—El jugador que consiga primero hacerse con 5.000 puntos es el ganador. Pueden jugar tantos jugadores como se quiera y se necesitan seis dados. Los jugadores van tirando los dados por turno y se puntúa del siguiente modo:

— *Un 5 vale 100.*
— *Un 2 vale 50.*
— *Tres 5 tirados en la misma jugada valen 1.000.*
— *Tres 2 en la misma tirada valen 500.*
— *Tres del mismo palo (salvo 2 y 5) valen 100 veces su valor (es decir, tres 1 valen 100 puntos, y tres 6 valen 600).*
— *Si en una sola tirada se consigue la secuencia 1, 2, 3, 4 y 5 se consiguen 1.500 puntos.*
— *2, 3, 4, 5 y 6 en una sola tirada valen 2.000 puntos.*

Observe que los 4 y los 6 no puntúan en este juego salvo como parte de una secuencia. Una vez que ha tirado, el jugador puede volver a tirar, si quiere, los dados que no puntúan. Sin embargo, si en la siguiente tirada no gana ningún punto, no podrá tirar más en esa baza y se le cuenta como si hubiera conseguido en total cero puntos.

Mientras se consigan puntos, se pueden seguir tirando los dados «sobrantes» (con seis y cuatro).

Este juego recompensa al jugador lento y cuidadoso con una buena puntuación, pero no le da ninguna posibilidad contra el jugador atreviso que decide arriesgarse a intentar conseguir una puntuación muy alta en una sola tirada.

JUEGOS DE CARTAS.—Una vez que se desarrolla una rutina para que el ordenador pueda mantener, barajar y manejar las cartas de un modo razonable (y una rutina de este tipo se encuentra en «PROBOSCIDEO»), se tiene ya la estructura precisa para desarrollar toda una multitud de juegos. Cualquier libro sobre el tema ofrece muchas más ideas de las que uno puede utilizar. Aquí van unas cuantas para empezar:

TRENTE ET QUARANTE.—Este juego, que nació en el siglo XVII en Europa, se conoce también con el nombre de ROUGE ET NOIR. En la versión que se juega en los casinos se utilizan seis barajas, pero se puede jugar utilizando simplemente una. Las cartas se reparten en dos filas examinándose el total de cada fila después de añadir una nueva carta. Las jotas, los reyes y las reinas cuentan diez y el resto de las cartas (incluyendo el as) cuentan su valor. En el momento en que el total de una fila sea igual o sobrepase a 31 se dejan de añadir cartas a esa fila. Sin embargo, el jugador que reparte sigue añadiendo cartas a la otra fila hasta que ésta también llegue o sobrepase 31. La primera fila se llama «Black» independientemente de las cartas que la compongan, y la segunda se denomina «red».

Los jugadores apuestan a cuál de las dos filas se aproximará más a 31. También pueden apostar a que el color de la *primera* carta repartida será o no igual al nombre del color de la fila ganadora. Todas las apuestas se hacen contra la banca y los ganadores reciben la cantidad apostada más la misma cantidad (uno a uno). Si las dos filas suman 31, la banca se queda con la mitad de la apuesta y devuelve el resto al jugador.

CARD CRAPS.— Este juego se ideó para sustituir a los dados al prohibirse éstos en algunos estados de Estados Unidos. Se utiliza una baraja especial de 48 cartas hecha utilizando los ases, doses, treses, cuatros, cincos y seises de otras dos barajas. Las reglas son las mismas

que las que se siguen en un juego de dados con el SIETE/ONCE que se incluye en este libro.

En su versión simplificada, eliminando el complicado sistema de apuestas, se juega del modo siguiente:

— Un total de 2, 3 ó 12 es «crap» y el jugador que reparte, la mano, pierde y pasa las cartas al siguiente jugador para que siga él levantando cartas.

— Un total de 7 u 11 es un «natural» y el jugador gana.

— Cualquier otro número (4, 5, 6, 8, 9 ó 10) es el «punto» del jugador y éste ha de continuar sacando cartas hasta que le vuelva a salir ese número, momento en que el jugador gana el juego. Sin embargo, si le sale un 7 antes de que le salga el número que busca, el jugador pierde la jugada.

THREE CARD BRAG.— Este juego fue el precursor del póker y se juega con tres cartas, en vez de las cinco del póker. Estas son las combinaciones que se pueden hacer clasificadas de las más altas a las más bajas:

— *«pryle», tres cartas del mismo tipo (p. ej., tres jotas...);*
— *«on a bike», una secuencia de tres cartas del mismo palo;*
— *«run», tres cartas seguidas;*
— *«flush», tres cartas del mismo palo;*
— *«pareja», dos cartas del mismo tipo;*
— *«la carta más alta».*

Se utiliza una baraja normal (de póker) y los ases son los que más valen, excepto en «on a bike» y «run» en donde 3, 2, as, vence a as, rey, reina. Este juego se puede modificar dando a cada jugador siete cartas en vez de tres; el jugador divide éstas en dos grupos de tres cartas a su elección o también se pueden dar nueve cartas en cuyo caso se crean tres cartas.

AS-DOS-JOTA.—Es este un sencillo juego en el que la mano, el jugador que reparte, o levanta, las cartas, lleva una gran ventaja. En este juego, los jugadores apuestan sobre la posibilidad de que la siguiente carta levantada será un as, un dos o una jota.

QUITA Y PON.—Este juego de apuestas usa una especie de peón o peonza de ocho lados que los jugadores hacen bailar o girar. Al comienzo del juego todos los jugadores tienen 10 fichas y el montón 30. Los jugadores hacen girar la peonza por turno y, siguiendo las instrucciones que aparecen en ésta, ponen o quitan fichas del montón. Cuando un jugador se queda sin fichas tiene que retirarse terminando el juego cuando ya no quedan fichas en el montón. Las instrucciones escritas en los lados de la peonza son:

— *Pon una.*
— *Quita una.*
— *Pon tres.*
— *Quita tres.*
— *Pon cuatro.*
— *Quita cuatro.*
— *Pon todas.*
— *Quita todas.*

BOULE.— Esta es una forma simplificada de la ruleta que apareció en el siglo XVIII. Se suele encontrar en los casinos europeos. La rueda está dividida en 18 segmentos, numerados de 1 a 9... por duplicado. El jugador apuesta al número ganador o a si será rojo o negro, par o impar, alto o bajo. Excepto los cincos de la rueda que suelen ser amarillos, los rojos y los negros alternan, de modo que hay, por ejemplo, un seis rojo y un seis negro, un nueve rojo y un nueve negro y así sucesivamente. Un número alto es un número por encima de cinco y los bajos serán los que estén por debajo de éste. Si un jugador apuesta por un número y sale este número, la banca le pagará siete a uno, en el caso de que haya apostado a una «característica» de ese número, es decir, par o impar, rojo o negro, alto o bajo, etc., se la paga uno a uno.

BIBLIOGRAFIA

Como quizá ya habrá podido darse cuenta el lector, es para mí muy importante el leer todos los libros sobre juegos que caen en mis manos, ya sean éstos sobre juegos con ordenadores o sobre los «anticuados» juegos de «humanos contra humanos». Siempre que se trate de un libro sobre juegos tengo una necesidad imperiosa de verlo.

Los libros de juegos son excepcionales para sacar ideas. Pruebe el lector a leer un solo juego y seguro que ya tendrá la semilla de un posible programa. Y esto no quiere decir que el programa acabe siempre siendo igual que el juego que ha leído. Muchos de los juegos para ordenadores empezaron siendo un simple intento de informatizar un juego tradicional y terminaron siendo un juego totalmente diferente, pero no por ello menos divertido.

Espero que alguno de los libros que cito a continuación puedan serle tan valiosos como lo han sido para mí:

Discovering old Board Games, R. C. Bell (Shire Publications, Aylesbury, UK, 1980).

Discovering Dice and Dominoes, R. C. Bell (Shire Publications, Aylesbury, UK, 1980).

Dice Games new and old, William E. Tredd (The Oleander Press, London and New York, 1981).

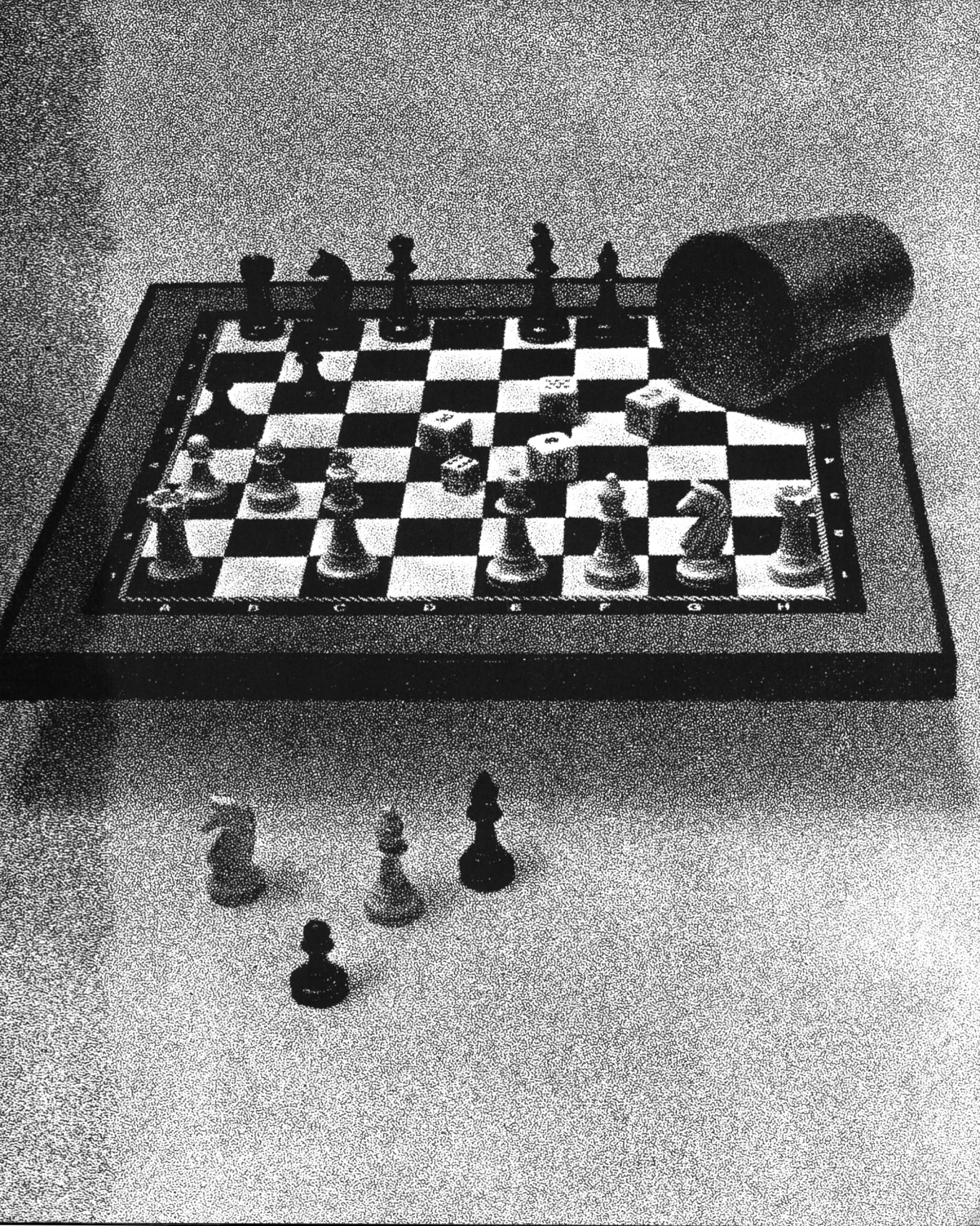
Game Playing with Basic, Donald D. Spencer (Hayden Book Co., Inc., Rochelle Park, New Jersey, 1977).

Games Ancient and Oriental and how to Play them, Edward Falkener (Dovar Publications, Inc., New York, 1961, a reprint of an original work published by Longmans, Green and Co. in 1982).

Basic Computer Games, Ed. David H. Ahl (Creative Computing Press, Morristown, New Jersey, 1978).
More Basic Computer Games, Ed. David H. Ahl (Creative Computing Press, Morristown, New Jersey, 1979).
26 Basic Programs for your Micro, Derrick Daines (Newnes Technical Books, Butterworths & Co., London and Boston, 1982).
Table Games of Georgian and Victorian Days, F. R. B. Whitehouse (Priory Press, Royston, Herts., UK, 1971).
Pet Fun and Games, Ron Jeffries and Glen Fisher (Osborne/McGraw-Hill, Berkeley, California, 1981).
Computer Games for Businesses, Schools and Homes, J. Victor Nahigian and William S. Hodges (Whinthrop Publishers, Inc., Cambridge, Massachusetts, 1979).
Games you Make and Play, Pia Hsiao, Neil Lorimer and Nick Williams (Macdonald and Jane's, London, 1975).
Waht to do After you hit Return, People's Computer Company (Hayden Book Co., Inc., Rochelle Park, New Jersey, 1980).
Everyman's Indoor Games, Gyles Brandreth (Dent, Everyman's Library, London, Melbourne and Toronto, 1982).
A Player's Guide to Table Games, John Jackson (Stackpole Books, Harrisburg, Pennsylvania, 1975).
Games for two, John Wasley (Proteus, London and New York, 1981).
The Addison-Wesley Book of Apple Computer Software 1982, Jeffrey Stanton and John Dickey, editors (The Book Company, Lawndale, California, 1982).
Dicing with Dragons, Ian Livingstone (Routledge & Kegan Paul, London, Melbourne and Henley, 1982).
Fantasy Role Playing Games, J. Eric Holmes (Hippocrene Books Inc., New York, 1981).

Libros sobre ajedrez de Computer Science Press:

Chess and Computer, David Levy (1976).
More Chess and Computers, David Levy and Monroe Newborn (1980).
All About Chess and Computers, David Levy and Monroe Newborn (1983).
(Computer Science Press, Rockville, Maryland).


A B C D E F G H

GLOSARIO

ACUMULADOR.—Parte de la unidad aritmético-lógica del ordenador que almacena los resultados intermedios.

ALGEBRA BOOLEANA.—El álgebra del mecanismo de toma de decisiones y lógica que desarrolló el matemático inglés George Boole. De ella depende la habilidad que muestra el ordenador a la hora de tomar decisiones.

ALGORITMO.—La secuencia de pasos dados en la solución de un problema.

ALFANUMERICO.—Se utiliza generalmente para describir el teclado que contiene teclas alfabéticas y numéricas, en comparación con el teclado numérico que sólo tiene teclas para los números del uno al nueve.

APL.—Siglas de Automatic Programming Language, un lenguaje desarrollado por Iverson a principios de los años sesenta que apoya una amplia serie de operadores y estructuras de datos. Usa un conjunto de caracteres no estándar.

ARCHIVO («File»).—Grupo de informaciones relacionadas entre sí que se procesan juntos, tales como el archivo de un estudiante, etc.

ASCII.—Siglas de «American Standard Code for Information Exchange». Se trata de un código casi universal para caracteres, números y símbolos.

BASE DE DATOS.—Colección de datos organizada de modo que el ordenador pueda acceder rápidamente a ella.

BASIC.—Siglas de «Beginner's All-purpose Symbolic Instruction Code»; es el lenguaje que se usa generalmente en los microordenadores. Es fácil de aprender, ya que la mayoría de las instrucciones son muy parecidas al inglés.

BAUDIO.—Unidad para medir la velocidad de transmisión de los datos. Normalmente representa el número de «bits» (unidades discretas de información) por segundo.

«BENCHMARK».—(Véase «Punto de referencia».)

BINARIO.—Sistema de numeración en el que hay sólo dos símbolos, 0 y 1 (en oposición al sistema decimal ordinario en el que hay diez símbolos, 0, 1, 2, 3, 4, 5, 6, 7, 8, y 9). Los ordenadores «piensan» en términos binarios.

«BOOTSTRAP».—Primer programa que se ejecuta tras la conexión del ordenador y que permite a éste aceptar y comprender otros programas.

«BUFFER».—(Véase «Regulador».)

«BUG».—Error en un programa.

BUS.—Grupo de conexiones eléctricas usadas para unir un ordenador a otro mecanismo auxiliar o a otro ordenador.

«BYTE».—El grupo más pequeño de «bits» (véase «bit») que forma una palabra en el ordenador. Generalmente un ordenador se describe por sus «bits», ya sea de «ocho bits» o de «dieciséis bits» lo que significa que la palabra consiste en una combinación de ocho o de dieciséis ceros o unos.

CAMPO.—Colección de caracteres que forman un grupo distinto, como un código de identificación, un nombre o una fecha; generalmente un campo forma parte de una información.

CIRCUITO PRINCIPAL («Motherboard»).—Unidad que está equipada de modo que se le puedan añadir al ordenador circuitos adicionales proporcionándole así características de las que carece ese modelo estándar (tales como gráficos de alta resolución o control de robot).

CODIGO DE CARACTERES.—Es el número ASCII (véase ASCII) que se refiere a un símbolo en particular, tales como 32 para un espacio o 65 para la letra «A».

COBOL.—Siglas de Common Business Orientated Language, un lenguaje de programación estándar, muy parecido al inglés, que se utiliza fundamentalmente en el mundo de los negocios.

COMPARTIDO.—Se utiliza este término cuando se da el caso de un gran número de usuarios, cada uno en un terminal distinto, que utilizan el mismo ordenador, el cual divide su tiempo entre los diferentes usuarios de modo que da la impresión de que les está dedicando «toda su atención» por separado.

COMPILADOR.—Programa que pasa otro programa escrito en un

lenguaje de alto nivel (parecido al humano) al lenguaje de la máquina de modo que ésta lo entienda directamente.

CONCATENAR.—Se llama concatenación a la suma de dos series.

CP/M.—Siglas de Control Program/Microcomputer; se trata de un sistema universal de operación por medio de disco desarrollado y puesto en el mercado por Digital Research, en Pacific Grove, California.

DATOS.—Término general para la información procesada por un ordenador.

«DEBUG».—Quitar los errores («bugs») de un programa.

DIAGRAMA DE FLUJO («Flowchart»).—Trazado escrito de la estructura y curso de un programa en el que se utilizan formas diferentes, como un rectángulo con los lados inclinados para indicar un ordenador en acción, un diamante para las decisiones tomadas por el ordenador. Normalmente, antes de introducir una sola línea del programa en el ordenador se suele hacer el gráfico de éste.

DIRECCION.—Número que se refiere a la posición, generalmente en la memoria del ordenador, en la que se almacena la información.

DISCO.—Un medio de almacenaje magnético (más adelante descrito como «disco duro» o «disco flexible») utilizado para almacenar la información del ordenador y los programas. Estos discos se parecen hasta cierto punto a los discos de música de 45 rpm y miden generalmente ocho, cinco y un cuarto o tres pulgadas de diámetro. En algunos sistemas también existen «microdiscos» más pequeños todavía.

DISCO FLEXIBLE.—Ver disco.

DOBLE DENSIDAD.—Adjetivo con el que se describe a los discos grabados utilizando una técnica especial que, como su nombre sugiere, duplica la cantidad de almacenaje que el disco proporciona.

DOCUMENTACION.—Las instrucciones y explicaciones escritas que acompañan a un programa.

DOS.—Siglas de «Disk Operating System»: el programa que permite que el ordenador controle un sistema de disco.

ENSAMBLADOR («Assembler»).—Es un programa que convierte otro programa escrito en lenguaje de ensamble («assembly language») en un código que el microprocesador puede ejecutar directamente.

ENTRADA (INPUT).—Toda la información que se introduce en un programa durante su ejecución.

EPROM.—Siglas de «Erasable Programmable Read Only Memory», un mecanismo que contiene información bajo una forma semi

permanente y que exige una exposición a la luz ultravioleta en el caso de querer borrar sus contenidos.

ESTRUCTURA PRINCIPAL (Mainframe).—Es el término utilizado en el caso de ordenadores «gigantes» como el IBM 307. Los ordenadores se clasifican asimismo como miniordenadores y microordenadores (el tipo de ordenador que posiblemente utilice el lector).

«FIRMWARE».—Los componentes físicos, la maquinaria de un sistema de ordenador se denominan «hardware»; los programas, en forma legible para la máquina ya sean en discos o cassettes, se llaman «software» y los programas que funcionan dentro del ordenador relacionados íntimamente con el «hardware» se denominan «firmware». El «firmware» puede alterarse, hasta cierto punto, por medio del «software».

FORTRAN.—Un lenguaje de ordenador de alto nivel que se usa generalmente para el trabajo científico (siglas de FORmula TRANslation).

«GATE».—«Componente» del ordenador que toma decisiones permitiendo que el circuito vaya en una u otra dirección dependiendo de las condiciones que ha de satisfacer.

GIGO.—Siglas de «Garbage In Garbage Out» con lo que se quiere indicar que si se introduce «basura» («garbage»), es decir, datos erróneos, el resultado del procesado de tales datos será también «basura».

GLOBAL.—El conjunto de condiciones que afectan a todo un programa se denominan «global», opuesto a «local» en el caso de que no afecten a todo un programa.

«HARD COPY».—La información que el impresor vierte en papel.

«HARDWARE».—Se denomina «Hardware» a la maquinaria física del ordenador (véase «software y «firmware»).

HEXADECIMAL.—Un sistema de contar muy apreciado por los programadores de código de máquina ya que está en estrecha relación con el método de almacenaje utilizado por los ordenadores; está basado en el número 16 en vez del 10, que es el número en el que se basa nuestro sistema de contar ordinario.

IMPRESOR DE LINEA.—Impresor que imprime una línea completa de caracteres de una sola vez.

IMPRESOR DE MATRIZ DE PUNTOS («Dot-Matrix Printer»).—Impresor que forma las letras y símbolos con puntitos, normalmente en una cuadrícula de ocho por ocho o de siete por cinco.

IMPRESOR TERMICO.—Mecanismo que imprime el «Output» del ordenador en un papel sensible al calor. Aunque los impresores térmicos son más silenciosos que los otros su salida «Output» no siempre es fácil de leer ni tampoco es fácil de almacenar el papel utilizado.

INTERPRETE.—Convierte el programa de alto nivel («comprensible por el humano») en un programa comprensible para el ordenador.

KILOBYTE.—Unidad de medida del lenguaje; un kilobyte (generalmente abreviado en K) equivale a 1024 «bits» (ver «bit»).

LENGUAJE DE ALTO NIVEL.—Lenguajes de programación parecidos a la lengua del usuario (generalmente al inglés). Los lenguajes de bajo nivel están más cercanos a los lenguajes comprensibles para el ordenador. Como los lenguajes de alto nivel han de pasarse a una forma comprensible por el ordenador antes de poder procesarlos, su ejecución es más lenta que la de los de bajo nivel.

LENGUAJE DE MAQUINA.—Es el escalón inferior a un lenguaje de bajo nivel; este es el lenguaje que el ordenador entiende directamente.

MEMORIA.—El mecanismo o mecanismos utilizados por el ordenador para guardar la información y los programas que se procesan así como el conjunto de instrucciones que le indican a éste el modo de llevar a cabo el programa. Básicamente existen dos tipos de memoria (ver RAM y ROM).

MEMORIA DINAMICA.—Memoria que ha de recargarse continuamente con el fin de que mantenga sus contenidos.

MEMORIA VOLATIL.—Un mecanismo de memoria que pierde sus contenidos cuando se le corta la corriente (ver «Memoria», «Refresco», ROM y RAM).

MICROPROCESADOR.—El «Chip» que hace de corazón del ordenador. Es la parte «pensante» de éste.

MODEM.—Siglas de «MOdulator/DEModulator, mecanismo que permite que el ordenador se comunique con otro por medio del teléfono.

MONITOR.—*a)* una pantalla de televisión usada como unidad de «display» de un ordenador; carece de mandos de ajuste; *b)* la información que lleva el ordenador y que le permite comprender y ejecutar las instrucciones del programa.

«MOUSE».—Unidad de control, ligeramente más pequeña que un paquete de tabaco, que rueda sobre la mesa y mueve en sentido paralelo un cursor sobre la pantalla con el fin de seleccionar las

diferentes opciones y tomar decisiones dentro del programa. Los «mouses» funcionan ya sea por la acción de sus ruedas o leyendo una cuadrícula en la superficie sobre la que se mueven.

OCTAL.—Un sistema de numeración basado en el ocho (utilizando los números 0, 1, 2, 3, 4, 5, 6 y 7).

«ON LINE».—Mecanismo que está bajo el control directo del ordenador.

PASCAL.—Un lenguaje de alto nivel desarrollado a finales de los años sesenta por Niklaus Wirth y que anima a una programación disciplinada y estructurada.

PILOT.—Lenguaje de nivel alto que se utiliza generalmente para desarrollar los programas de ordenador utilizados en educación.

PONER A CERO («Reset»).—Señal que vuelve al ordenador al punto en el que estaba al encenderlo.

«PORT».—Punto de entrada o salida a través del que se transfieren los datos en un ordenador.

PROCESADOR DE TEXTO («Word processor»).—Es un ordenador dedicado (o un ordenador que opere un programa de proceso de textos), que da acceso a una «máquina de escribir inteligente» con una amplia gama de posibilidades de ajuste y corrección.

PROGRAMA.—Serie de instrucciones que sigue el ordenador para llevar a acabo una tarea determinada.

PUNTO DE REFERENCIA («Benchmark»).—Prueba que se realiza para medir algún aspecto del funcionamiento de un ordenador que se puede comparar al resultado de ejecutar una prueba similar en otro ordenador.

RAM.—Siglas de «Random Access Memory»; es la memoria que permite al ordenador mantener el programa en curso de ejecución. Los contenidos del RAM pueden cambiarse mientras que los contenidos del ROM no pueden cambiarse bajo el control del «software».

RED («Network»).—Grupo de ordenadores que funcionan juntos.

REGULADOR («Buffer»).—Mecanismo de almacenaje que mantienen la entrada a partir de un mecanismo tal como una tecla, tecla que va soltando con la frecuencia que le dicta el ordenador.

REFRESCO.—Las memorias dinámicas (ver «memoria») han de recibir irrupciones de electricidad periódicas con el fin de que mantengan sus contenidos. La señal que recuerda a la memoria cuáles son sus contenidos se llama señal de refresco.

REGISTRO.—Punto de la memoria del ordenador que contiene datos de frecuente uso.

ROM.—Véase RAM.

RS-232.—Norma de conexión estándar que conecta un Modem o el equipo asociado a la terminal a un ordenador.

S-100 BUS.—Se trata también de una norma de conexión estándar (véase RS-232) hecha con 100 líneas de comunicación común paralelas que se usan para conectar los circuitos dentro de los microordenadores.

SALIDA (OUTPUT).—Todos los datos producidos por el ordenador mientras está procesando, ya aparezcan estos datos en la pantalla o los vierta al impresor o los use internamente.

SNOBOL.—Lenguaje de alto nivel desarrollado por los laboratorios Bell que utiliza el reconocimiento de esquemas o dibujos y manipulación de las series o cadenas («strings»).

SEÑALIZADOR («Flag»).—Indicador interior del programa que da información sobre una condición determinada.

SISTEMA OPERATIVO («Operating System»).—Es un programa que controla el funcionamiento general del ordenador y sus periféricos.

SISTEMA «LLAVE EN MANO».—Sistema (generalmente para uso en el mundo de los negocios) que está preparado para ejecutar y sólo necesita que «se apriete una tecla» para que funcione.

«SOFTWARE» (logical).—Es el programa que ejecuta el ordenador (véase «Firmware»).

«SOFTWARE» APLICADO.—Programas escritos para la realización de tareas especiales, como procesado de palabras o listas de correspondencia.

«SOFTWARE» DE SISTEMAS.—Secciones de códigos que llevan a cabo tareas administrativas dentro del ordenador o ayudan en la escritura de otros programas, pero que no se usan para realizar la tarea que se quiere que realice el ordenador.

SINTAXIS.—Como en las lenguas humanas, la sintaxis es el conjunto de reglas estructurales que gobiernan el uso de un lenguaje de ordenador.

SUBRUTINA.—Parte del código, o programa, a la que se llama un número determinado de veces durante la ejecución del programa.

TECLADO HEX («Hex Pad»).—Teclado, parecido al de una calculadora, que se usa para introducir directamente los números hexadecimales.

TECLADO NUMERICO («Numeric Pad»).—Mecanismo utilizado para introducir información numérica en el ordenador, parecido a una calculadora.

TIEMPO REAL.—Cuando una acción realizada en el ordenador progresa paralelamente al tiempo del «mundo real», se dice que la acción ocurre en tiempo real. Un ejemplo sería un programa que mostrara el desarrollo de una colonia de bacterias que se reprodujeran con el mismo ritmo de crecimiento que lo haría una colonia real. Ultimamente han aparecido muchos juegos que requieren reacciones en tiempo real. La mayoría de los juegos de las máquinas de los salones recreativos transcurren en tiempo real.

OTROS TITULOS PUBLICADOS EN ANAYA MULTIMEDIA

EL LIBRO DEL BASIC - Rodnay Zaks.
TU PRIMER LIBRO DEL ZX SPECTRUM - J. Dewhirst & R. Tennison.
PROGRAMACION EN BASIC: UN METODO PRACTICO - H. Dachslager/M. Hayashi/R. Zucker.
BITS Y BYTES: INICIACION A LA INFORMATICA - Rachelle S. Heller & C. Dianne Martin.
EL ORDENADOR EN EL AULA - Egidio Pentiraro.
ASTRONOMIA: EL UNIVERSO EN TU ORDENADOR (ZX SPECTRUM) - Maurice Gavin.
EL ORDENADOR Y TUS HIJOS - Ray Hammond.
EL ORDENADOR PERSONAL: COMO ELEGIRLO Y UTILIZARLO - Arnoldo Cavalcoli.

Editados por:
EDICIONES ANAYA MULTIMEDIA, S. A.
Villafranca, 22
28028 MADRID

Revistas especializadas

EL ORDENADOR PERSONAL
Ferraz, 11, 3.º
28008 MADRID

MICROS
Ediciones Arcadia
Víctor de la Serna, 4
28016 MADRID

TU MICRO
Avenida Alfonso XIII, 141
28016 MADRID

ORDENADOR POPULAR
Ediciones y Suscripciones, S. A.
Jerez, 3
28016 MADRID

INFORMATICA TEST
Haymarket, S. A.
Travesera de Gracia, 17, 21
BARCELONA 21

CHIP
Ediciones Arcadia
Víctor de la Serna, 4
28016 MADRID

ZX
Apartado 784
MADRID

NOTAS

Cara 1

0 - 12 Futbol
13 - 22 Globo
23 - 39 Partida
30 - 44 Aluzinage-I
46 - 52 ~~Caracteres~~ Cracteres

NOTAS

EL LIBRO GIGANTE DE LOS JUEGOS PARA ORDENADOR es una recopilación cuidadosamente escogida de más de 40 juegos para ordenador agrupados por temas: Juegos de Mesa, Juegos de Aventura, Simulaciones, Juegos de Dados, Inteligencia Artificial, Pura diversión, Juegos espaciales, Juegos de Concentración. El libro, además de proveer material para muchísimas horas de diversión con el ordenador, explica detalladamente las bases del diseño de juegos, a lo que dedica un capítulo en especial, comentando en cada tipo de juegos los algoritmos en los que se basan, cómo están expresados en el programa y cómo se pueden mejorar. Los listados de los programas son utilizables en cualquier microordenador programable en BASIC.

ANAYA MULTIMEDIA